

AD-A065 136

CHARLES STARK DRAPER LAB INC CAMBRIDGE MA

F/G 1/3

AN INTEGRATED FAULT-TOLERANT AVIONICS SYSTEM CONCEPT FOR ADVANC--ETC(U)

FEB 79

N00019-78-C-0572

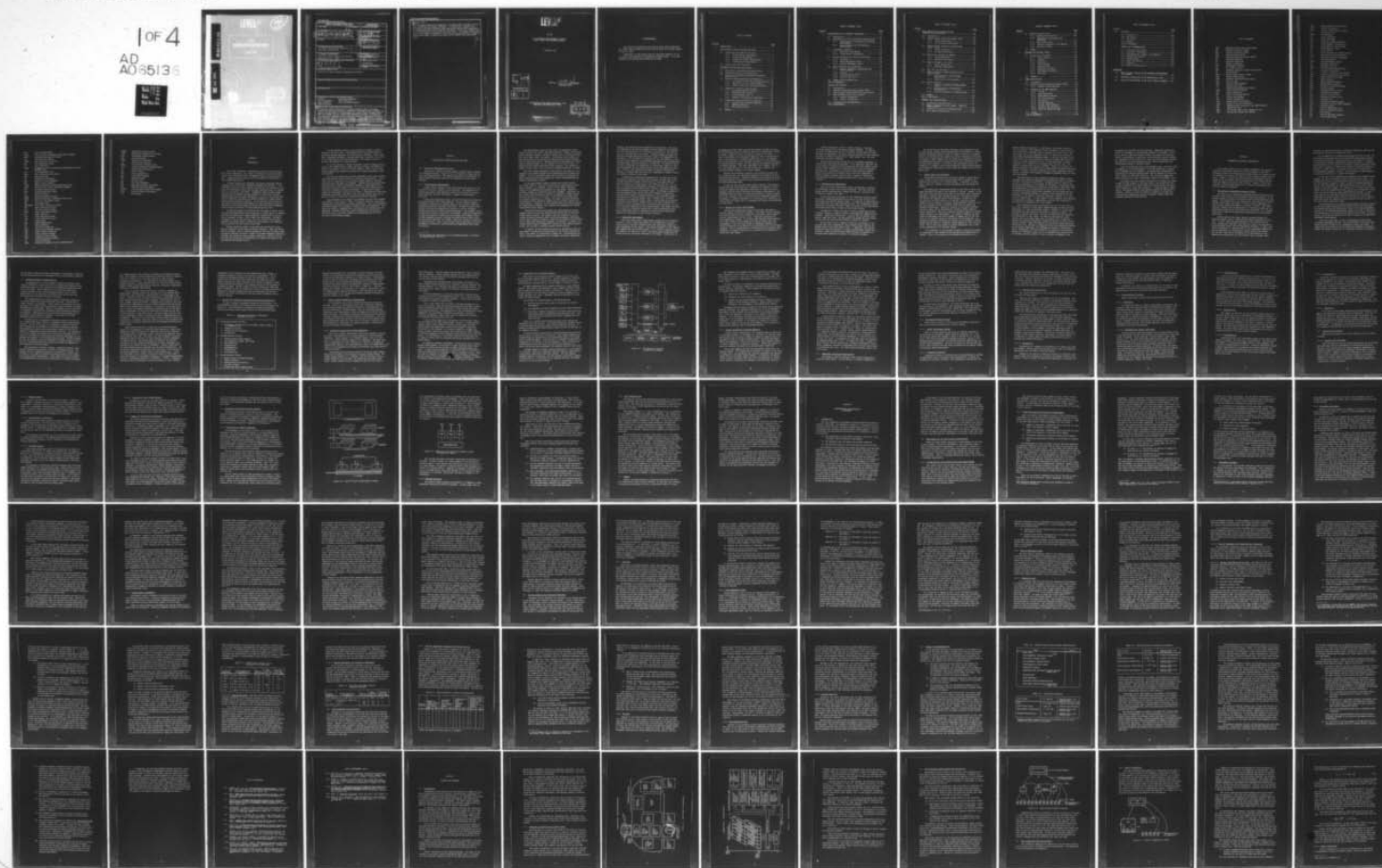
UNCLASSIFIED

R-1226

NL

1 OF 4

AD  
A065136





DDC FILE COPY

AD A0 651 36


LEVEL



REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) <b>AN INTEGRATED FAULT-TOLERANT AVIONICS SYSTEM CONCEPT FOR ADVANCED AIRCRAFT</b>		5. TYPE OF REPORT & PERIOD COVERED <b>Final Report</b>
7. AUTHOR(s)		6. PERFORMING ORG. REPORT NUMBER <b>R-1226</b>
9. PERFORMING ORGANIZATION NAME AND ADDRESS <b>The Charles Stark Draper Laboratory, Inc. Cambridge, Massachusetts</b>		8. CONTRACT OR GRANT NUMBER(s) <b>N00019-78-C-0572</b>
11. CONTROLLING OFFICE NAME AND ADDRESS <b>Department of the Navy Naval Air Systems Command Code PMA-2693 Washington, D.C. 20361</b>		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) <b>325 P1</b>		12. REPORT DATE <b>1 February 1979</b>
16. DISTRIBUTION STATEMENT (of this Report) <b>Approved for public release; distribution unlimited.</b>		13. NUMBER OF PAGES <b>328</b>
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		15. SECURITY CLASS. (of this report) <b>UNCLASSIFIED</b>
18. SUPPLEMENTARY NOTES		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) <div style="display: flex; justify-content: space-between;"> <div> <b>Avionics</b>  <b>System Integration</b>  <b>Fault Tolerance</b>  <b>Damage Tolerance</b> </div> <div> <b>Redundancy Management</b>  <b>Fault Detection</b>  <b>Fault Tolerant Computers</b> </div> </div>		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) <p>A conceptual baseline design for a highly integrated fault- and damage-tolerant avionics architecture is presented. The architecture is generic in nature, and applicable to a broad range of aircraft types; including CTOL, VTOL, and V/STOL; and all classes from supersonic fighters to transports. The architecture embodies pools of modular resources, configured to flexibly serve required functions on a priority basis. By including system elements which can serve multiple functions and taking maximum advantage of systematic</p>		

Abstract (Cont.)

fault-tolerance methods and procedures, the design tends to minimize replication of elements and overall complexity. In concert with logistics and maintenance procedures designed around the pooled modular element approach, the architecture can provide required performance, reliability, damage tolerance, and availability at minimum life-cycle costs. Its inherent flexibility allows it to readily incorporate a wide variety of mission-specific elements and to easily adapt to growth and change as new elements and requirements arise.





# LEVEL II

R-1226

AN INTEGRATED FAULT-TOLERANT AVIONICS  
SYSTEM CONCEPT FOR ADVANCED AIRCRAFT

1 February 1979

ACCESSION FOR		
DTIS	White Section	<input checked="" type="checkbox"/>
DD	Buff Section	<input type="checkbox"/>
UNANNOUNCED		<input type="checkbox"/>
JUSTIFICATION		
BY		
DISTRIBUTION/AVAILABILITY CODE		
Dist.	AVAIL.	AND/OR SPECIAL
A		

Approved:

*K. C. Sears*  
Department Head

The Charles Stark Draper Laboratory, Inc.  
Cambridge, Massachusetts 02139

DDC  
RECEIVED  
MAR 2 1979  
D

DISTRIBUTION STATEMENT A

Approved for public release;  
Distribution Unlimited

#### ACKNOWLEDGMENT

This report was prepared by The Charles Stark Draper Laboratory, Inc. under Contract N00019-78-C-0572 with the Naval Air Systems Command of the U.S. Navy.

Publication of this report does not constitute approval by the U.S. Navy of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

*Prepared for publication by the CSDL Publications Section.*



## TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
1 INTRODUCTION.....	1
2 CORE AVIONICS SYSTEM FUNCTIONS AND GOALS.....	3
2.1 Functional Requirements and Goals.....	3
2.1.1 Flight-Control Requirements.....	3
2.1.2 Navigation Requirements.....	5
2.1.3 Display and Control Requirements.....	6
2.1.4 Communications Requirements.....	7
2.2 General Goals and Procedures.....	8
3 INFORMATION-PROCESSING ARCHITECTURE.....	11
3.1 The Changing Information-Processing Problem.....	11
3.2 Technology Trends and Opportunities.....	13
3.3 Criteria for Assessing Information-Processing Architectures.....	15
3.4 Avionics System Computation Architectures.....	16
3.4.1 Dissimilar Primary and Secondary Resources..	16
3.4.2 Replication of the Primary Resource.....	18
3.4.3 Pooled, Dynamically Allocated Resources....	20
3.5 Comparison of Avionics Architectures.....	21
3.5.1 Performance-Related Criteria.....	22
3.5.2 Economy-Related Criteria.....	24
3.5.3 Safety-Related Criteria.....	26
3.5.4 Summary of Architecture Comparisons.....	28
3.6 Information System Architectural Baseline.....	29
3.6.1 Fault-Tolerant Computer Complexes.....	29
3.6.2 Embedded Computers.....	31
3.6.3 Data Communications.....	33
3.7 Summary.....	33



## TABLE OF CONTENTS (Cont.)

<u>Section</u>	<u>Page</u>
4 INSTRUMENTATION AND REDUNDANCY MANAGEMENT.....	35
4.1 Introduction.....	35
4.2 Requirements of the Core Avionics System Design.....	36
4.2.1 Flight-Control and Navigation Functional Requirements.....	36
4.2.2 Fault-Tolerance and Survivability Requirements.....	37
4.3 Instruments Available.....	39
4.3.1 Navigation Instruments.....	40
4.3.2 Flight-Control Instruments.....	42
4.4 Failure Detection and Isolation Techniques.....	46
4.4.1 Self-Test.....	47
4.4.2 Direct-Redundancy Tests.....	48
4.4.3 Analytic-Redundancy Tests.....	51
4.5 Major Tradeoffs and Conclusions.....	58
4.5.1 Inertial Components (Navigation and Flight Control).....	59
4.5.2 Air Data.....	61
4.5.3 Radio-Navigation Aids.....	62
4.5.4 Autoland Receivers.....	63
4.6 Summary and Recommendations.....	64
LIST OF REFERENCES.....	71
5 DISPLAYS AND CONTROLS.....	73
5.1 Introduction.....	73
5.2 Advanced Integrated Display System (AIDS).....	74
5.3 Fault-Tolerant System Network and Architecture.....	78
5.4 AIDS Integration With Core Avionics.....	79
5.4.1 Level-1 Integration.....	80
5.4.2 Level-2 Integration.....	82
5.5 Conclusions and Recommendations.....	86
LIST OF REFERENCES.....	87

## TABLE OF CONTENTS (Cont.)

<u>Section</u>	<u>Page</u>
6	RADIO NAVIGATION AND COMMUNICATIONS ALTERNATIVES/REQUIREMENTS.....88
6.1	Introduction.....88
6.2	NAVSTAR Global Positioning System (GPS).....88
6.2.1	System Description.....88
6.2.2	User Equipment Description.....90
6.3	Joint Tactical Information Distribution System (JTIDS).....91
6.3.1	System Description.....91
6.3.2	Equipment Description.....94
6.4	Tactical Air Navigation (TACAN).....95
6.4.1	System Description.....95
6.4.2	Equipment Description.....96
6.5	Identify Friend or Foe (IFF).....96
6.5.1	System Description.....96
6.5.2	Equipment Description.....96
6.6	UHF, VHF, and HF Radios.....97
6.7	Level-1 Approach to Radio Navigation and Communications.....97
6.7.1	Radio-Navigation Configuration Alternatives.....97
6.7.2	Communications Configuration.....101
6.8	Level-2 Approach.....105
6.8.1	Tactical Information Exchange System (TIES).....105
6.8.2	Radio-Navigation Configuration Alternatives.....110
6.8.3	Integrated Approach to Communications.....113
6.9	Summary.....116
	LIST OF REFERENCES.....120
7	INTERNAL DATA COMMUNICATIONS.....121
7.1	Introduction and Definition.....121
7.2	Highly Integrated Avionics System: Impact on Communications.....123
7.3	Limitation of Current Bus Architectures.....125
7.4	Basic Network Architecture.....128



## TABLE OF CONTENTS (Cont.)

<u>Section</u>	<u>Page</u>
7.5 Detailed Network Design.....	135
7.5.1 Communications Protocol.....	135
7.5.2 Transmission Technology—Link Technology.....	137
7.5.3 Node Construction.....	139
7.5.4 Terminal Attachment to the Network.....	141
7.5.5 Mechanical Design.....	142
7.6 Summary.....	143
8 SOFTWARE AND RESOURCE SIZING.....	144
8.1 Core-Avionics Functions.....	146
8.1.1 Flight Control.....	146
8.1.2 Guidance.....	147
8.1.3 Navigation.....	147
8.1.4 Crew Interface.....	147
8.1.5 Communications.....	148
8.1.6 Subsystem Processing.....	148
8.1.7 FTMP Operating System.....	150
8.1.8 Data Base.....	150
8.2 Summary.....	150
9 POWER DISTRIBUTION.....	157
9.1 Introduction.....	157
9.2 Advanced Aircraft Electrical System (AAES).....	157
9.2.1 Summary of AAES Features.....	159
9.3 Comments on the AAES Approach.....	162
9.3.1 270 VDC.....	162
9.3.2 Power Generators.....	162
9.3.3 Semiconductor Devices.....	163
9.3.4 DC to DC Converter.....	163
9.3.5 Contactors.....	163
9.3.6 EMI and Lightning.....	164
9.3.7 The MAP, AAES Interface.....	164
9.3.8 Grounding and Shielding.....	165
9.3.9 Power for SOSTEL and GPMS.....	167
9.4 Summary.....	167
LIST OF REFERENCES.....	168

## TABLE OF CONTENTS (Cont.)

<u>Section</u>	<u>Page</u>
10	PACKAGING.....169
10.1	Introduction.....169
10.2	MAP Program.....170
10.3	Discussion.....175
10.4	Summary.....179
	LIST OF REFERENCES.....181
11	SUMMARY AND RECOMMENDATIONS.....182
11.1	Functional Requirements.....182
11.2	Information Processing.....183
11.3	Instrumentation, Control, and Guidance.....186
11.4	Displays and Controls.....189
11.5	Communications.....190
11.6	Power Distribution.....192
11.7	Packaging.....193
11.8	General Considerations.....193

### APPENDICES

3-A	FTMP—A HIGHLY RELIABLE FAULT-TOLERANT MULTIPROCESSOR FOR AIRCRAFT.....195
6-A	FUNCTIONAL DESCRIPTION OF THE NAVSTAR GPS X-SET.....217
6-B	FUNCTIONAL DESCRIPTION OF THE JTIDS CLASS-II TERMINAL....283

## LIST OF ACRONYMS

AAES	Advanced Aircraft Electrical System
ADF	Automatic Direction Finder
ADM	Advanced Development Model
AEW	Airborne Early Warning
AIDS	Advanced Integrated Display System
AK	Altitude Kinematics
AM	Amplitude Modulation
AR	Analytic Redundancy
ASW	Antisubmarine Warfare
ATC	Air Traffic Control
ATR	Standard transport aircraft avionics box
BC	Bus Contactor
BFCs	Backup Flight Control System
BFS	Backup Flight System
BIED	Briefing Information Entry Device
BIT	Built-in Test
BITE	Built-in Test Equipment
C/A	Coarse/Acquisition
C <sup>3</sup>	Communication Command and Control
CCSK	Cyclic Code-Shift Keyed
CCU	Cable Control Unit
CCV	Control Configured Vehicle
C/D	Control and Display
CDU	Control Display Unit
CFA	Common Family Architecture
CILOP	Change in Lieu of Procurement
CNI	Communications, Navigation, and Identification
CRT	Cathode Ray Tube
CSDL	The Charles Stark Draper Laboratory, Inc.
CTOL	Conventional Takeoff and Landing



DEFT	Design Evaluation Flight Test
DFBW	Digital Fly By Wire
DFCS	Digital Flight Control System
DG	Directional Gyro
DMUX	Demultiplexer
DR	Direct Redundancy
DSB	Double-Side Band
ECM	Electronic Counter Measures
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
EMP	Electromagnetic Pulse
FC	Flight Control
FDI	Fault Detection and Identification
FDI	Fault Detection and Isolation
FDM	Frequency Division Multiplexed
FI	Fault Indicators
FLIR	Forward-Looking Infrared
FM	Frequency Modulation
FSK	Frequency Shift Key
FTMP	Fault-Tolerant Multiprocessor
GK	Geographic Kinematics
GN&C	Guidance, Navigation, and Control
GPMS	General-Purpose Multiplex System
GPS	Global Positioning System
HF	High Frequency
HIT	Hughes Improved Terminal
HMD	Helmet-Mounted Display
HSD	Horizontal Situation Display
HUD	Head-Up Display
ICP	Integrated Control Panels
IF	Intermediate Frequency
IFF	Identify Friend or Foe
IFF	Information Friend or Foe
IFU	Interface Unit
IHS	Information Handling System
IISA	Integrated Inertial Sensor Assembly
IM	Interface Module
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
I/O	Input/Output
IRA	Inertial Reference Assembly
IUS	Interim Upper Stage



JPO	Joint Program Office
JTIDS	Joint Tactical Information Distribution System
KOPS	Thousand Operations Per Second
LED	Light-Emitting Diodes
LLLTV	Low-Light-Level Television
LMC	Load Management Center
LO	Local Oscillator
LORAS	An Omnidirectional Low-Range Airspeed System by Pacer Systems, Inc.
LRU	Line Replaceable Unit
LSAD	Left Status Advisory Display
LSB	Lower-Side Band
LSI	Large-Scale Integration
MAP	Modular Avionics Packaging
MCS	Master Control Station
MIDER	Modular-Integrated Display Electronics Rack
MIRA	Multifunction Inertial Reference Assembly
MLS	Microwave Landing System
MPSK	Minimum Phase-Shift Keyed
MS	Monitor Stations
MTBF	Mean Time Between Failures
MSPRT	Modified Sequential Probability Ratio Test
NADC	Naval Air Development Center
NAV	Navigation grade
NAVTO LAND	Navy V/STOL capability program
NEC	Naval Electronics Center
NIC	New Installation Concept
NOSC	Naval Ocean Systems Center
PC	Power Controller
PGS	Power Generation System
PIN	P-type Intrinsic N-type
PR	Pseudo Range
PRN	Pseudo-Random Noise
RF	Radio Frequency
RK	Rotational Kinematics
RSAD	Right Status Advisory Display
RSG	Raster Signal Generation
SAD	Status Advisory Display
SAM	Standard Avionics Module
SEM	Standard Electronics Module
SI	Status Indicators
SIRU	Redundant Strapdown Inertial Reference Unit

SOSTEL	Solid-State Electric Logic
SPRT	Sequential Probability Ratio Test
SRLS	Short-Range Landing System
SSB	Single-Side Band
TACAN	Tactical Air Navigation
TD	Translational Dynamics
TDMA	Time Division Multiple Access
TIES	Tactical Information Exchange System
TK	Translational Kinematics
TOA	Time of Arrival
TSE	Total Squared Error
UHF	Ultra-High Frequency
USB	Upper-Side Band
VG	Vertical Gyro
VHF	Very High Frequency
VOR	Very High Frequency Omni Range
VSD	Vertical Situation Display
V/STOL	Vertical/Short Takeoff and Landing
VTOL	Vertical Takeoff and Landing
$\Delta R$	Delta Range



## SECTION 1

### INTRODUCTION

This report documents a three-month study effort by The Charles Stark Draper Laboratory, Inc. (CSDL) to establish a preliminary definition of an integrated fault- and damage-tolerant avionics architecture. Operational Navy aircraft of the mid 1990s are envisioned as the intended application.

To assure appropriate cognizance of ongoing technology efforts, information on current Navy technology programs was gathered. A comprehensive review of these programs, presented by Navy personnel at The Naval Air Development Center, was the most important source of information for this part of the effort. Subsequent contact between Draper Laboratory staff members and Navy personnel at NADC, NEC, and NOSC, provided additional detailed information. Visits to, and contact with, various industrial contractors provided information on those program elements being pursued by industry. To the greatest extent practical, the current and projected developments of these programs have been integrated into the fault-tolerant avionics architecture and recommendations on future directions of these programs are set forth.

An important aspect of the effort was the desire to define a generic avionics system which could be applied to a broad array of vehicles. The range of application includes supersonic fighters and attack aircraft, transport aircraft, CTOL, V/STOL, and helicopters. Because it is common to all of these aircraft, the core avionics system (which supplies the flight-control, navigation, pilot-display, and communications functions) was the focus of this study.

Another important element of the study was the ability of the design to support a broad range of mission functions. Thus, the design must be flexible, able to interface gracefully with many different types of mission elements, and able to supply the systems management functions necessary for effective overall system integration. Implicit here is the need for flexibility and adaptability to growth and change as new requirements and systems evolve.

A third important aspect of the study was to define a system to minimize life-cycle costs. Considerable effort was concentrated on minimizing system complexity and replication of elements. Fault tolerance, redundancy management, and real-time fault isolation to the level of line replaceable units were emphasized in the design in order to minimize maintenance and logistics costs.

Of all the vehicles to which the generic avionics system could be applied, the V/STOL tends to impose the most stringent requirements in terms of its impact on the core avionics design. Hence, in the study, special emphasis was placed on assuring that V/STOL requirements could be satisfied.

The avionics architecture concept presented in this report differs significantly from current practice in avionics system design. A very high level of integration of all avionics system elements, including flight-critical functions, is proposed. Although this architecture has the potential to achieve significant increases in performance while minimizing life-cycle costs; no definitive flight-test experience with this type of system is currently available. To make such a system viable for operational aircraft of the 1990s, a comprehensive flight-test demonstration program, initiated in the very near future, would be required in order to verify the integrated architecture approach.

Major elements of the system are addressed in each of the succeeding sections of this report. First, the overall functions, operational requirements, and goals of the design are defined. Then, the information-processing aspects are discussed, followed by the instrumentation. Displays and controls, communications, and internal data transmission are addressed, followed by software, power distribution, and packaging. The final section provides a summary and recommendations for future directions of Navy technology programs.



## SECTION 2

### CORE AVIONICS SYSTEM FUNCTIONS AND GOALS

#### 2.1 Functional Requirements and Goals

The core of a generic avionics system is defined here to support four primary functions: flight control\*, navigation, display and control, and communications. The system design goals to be attained in each of these functional areas will be examined in the following sections.

##### 2.1.1 Flight-Control Requirements

Flight control consists of the stabilization and control of the aircraft's attitude, velocity, and flight path. The flight-control system must assure a well behaved aircraft from the point of view of pilot handling, and must also support a number of path-guidance functions.

Future high-performance combat aircraft will make use of Control Configured Vehicle (CCV) design methods. The CCV approach relaxes traditional constraints on airframe design, such as static stability, allowing significant performance advantages. Automatic controls then provide the stability augmentation necessary to attain acceptable handling qualities and overall system performance. For example, V/STOL aircraft are, by their very nature, CCV designs. Stability augmentation of most CCV aircraft is flight-critical because direct manual control of these unstable aircraft is beyond the capabilities of most pilots. Vertical-flight operation of a V/STOL, under poor visibility conditions, is flight-critical. Thus, flight control must be provided at levels of performance, reliability, and survivability commensurate with flight criticality.

---

\* In this report the term avionics is interpreted broadly, to include the flight-control function.

The path-guidance functions that must be supported by flight control include fire control and weapon delivery, landing, and various mission-oriented automatic navigation modes (such as terrain-following and low-altitude flight over water). In addition, path guidance for fire control may involve certain CCV design features such as direct-lift and side-force control, fuselage pointing, and precision tracking using a tracking sensor, as (for example) a fire-control radar. As weapon delivery tasks become more highly automated, the interaction between flight-control and fire-control elements inevitably increases.

The trend toward automation of path-control functions is a consequence of the growing capability and sophistication of sensors and weapons. It is an increasing trend that is likely to accelerate as combat scenarios become more complex and overall weapons-system performance requirements increase.

As the path-guidance elements become more closely coupled with flight control, the flight criticality of these elements inevitably increases. For example, failure of the terrain-following system can be immediately catastrophic at high speed. Similarly, automatic landing is flight-critical near touchdown. Typically, the more tightly the path-control function is coupled to flight control, the more critical it is to flight. In the terrain-following task with precise navigation to a target, the radar, inertial-navigation, and possibly radio-navigation elements are all tightly coupled with flight control. Similarly, in the autoland task, the autoland sensors and communication elements, both airborne and shipboard, are closely coupled with flight control.

Traditional distinctions between the roles played by various elements within the avionics system become blurred as the functions become tightly coupled. For example, when the aircraft is in the terrain-following navigation mode, the entire aggregate, consisting of the radar, inertial navigator, flight-control sensors and actuators, and all associated computation elements, is involved. In effect, the flight-control system is expanded to include all the elements participating, and the entire aggregate becomes flight-critical.

To support the path-guidance functions at appropriate levels of performance, reliability and survivability, the avionics system must have a high degree of configuration flexibility. The system must be able to access an appropriate set of sensors, actuators, computation



elements, and associated data-communications facilities. The levels of redundancy of the various elements must be chosen to be consistent with the required levels of reliability, survivability, and performance for the functions to be served. Elements capable of supporting multiple functions should play multiple roles in the designated configuration. The collection of elements must be tied together so that effective fault detection and redundancy management can be accomplished by comparisons of diverse sources of information. An efficient partitioning of tasks must be established in hierarchical fashion so that the requirements for data communications within the system are realizable. Finally, the configuration capability must be sufficiently flexible to readily admit new functions, subsystems or modes of operation that may evolve in the future.

A particularly significant consequence of the need for various path-control functions (as previously outlined) is that there is not a sharp distinction between those elements of the system that are strictly devoted to flight control, and those that are not. If, in fact, a flight-control subsystem is defined to include all elements that participate in the flight-control function, the entire collection would encompass much of the avionics system. Furthermore, imposing a dichotomy between flight control and other elements is directly at odds with the concept of system integration, and inevitably leads to a proliferation of system elements and a reduced level of overall flexibility to growth and change. Thus, a ground rule of the generic integrated fault-tolerant system design is that the flight-control function is fully integrated with, and shares resources with, all the other functions supported by the avionics system. The design challenge is to create an information network and a power-distribution system within which this level of integration can be realized without degrading the reliability of this critical function.

#### 2.1.2 Navigation Requirements

There are three primary requirements imposed upon the navigation function. First, a basic requirement is the capability to locate a target within a tactical reference frame with sufficient accuracy to allow engagement of that target. The precise statement of this requirement depends upon the aircraft mission. In a close air-support mission, for example, the need is to locate a specified ground target with sufficient accuracy to deliver ordinance on the target. In contrast, an

antisubmarine warfare (ASW) aircraft must deploy a pattern of buoys, and continuously define its position relative to those buoys in order to locate a submarine within the acoustic range of the buoys. A second requirement on the navigation system is that it provide guidance to the vicinity of a desired landing site (e.g., air-capable ship or forward base), with sufficient accuracy to allow acquisition of landing guidance signals, and then provide the guidance information for precise landings. The third requirement is to support fire control and weapon delivery in the form of position, velocity, and attitude transfer alignment to standoff weapons.

Because the navigation function is only mission-critical and not flight-critical, its reliability need not be as high as for the flight-control function. Important exceptions are cases in which the navigation system is coupled into the flight-control function, as in the terrain-following mode.

As was the case for the flight-control function, the avionics system must have the flexibility to assemble the outputs of various navigation sensors and computation facilities in order to provide necessary navigation data consistent with mission requirements. This assemblage must be managed in hierarchical fashion to assure maximum use of all resources, and to provide efficient redundancy management. Furthermore, a primary requirement is a high degree of flexibility to allow growth and change, and to accommodate new sensors and subsystems as they evolve.

### 2.1.3 Display and Control Requirements

The avionics system must support both flight- and mission-critical display and control functions. The flight-critical displays and controls must be highly reliable and survivable, while mission-critical requirements are much lower. These two categories are quite compatible, however, in the sense that the data rates required for flight control are low compared to typical mission data-rate requirements. Thus, the high levels of redundancy necessary for flight criticality can be obtained by multifunction use of the mission displays. This is accomplished by curtailing mission functions (when necessary) to support flight-critical functions in the presence of faults and local damage.



Flight displays may include a head-up display, a vertical-situation display, and a horizontal-situation display. In addition to these, the system must be able to display various types of status and warning information. Provision must also be made to input commands and data (in terms of control stick, rudder pedals, etc.; for mode switching; and from a keyboard).

Mission displays must provide for the increasing complement of tactical data that must be presented. It is especially important that a high degree of flexibility be incorporated into this display system so that future systems can be supported without requiring extensive modifications. Furthermore, the mission-related display elements should be included as an integral part of the total complement of display elements, thereby allowing them to support flight-control functions when faults or damage eliminate one or more of the flight displays.

#### 2.1.4 Communications Requirements

The core avionics system must support transmission, reception, and processing of both voice and data messages. The functions supported include communications, command and control, navigation and information friend or foe (IFF).

In the time frame envisioned for the integrated avionics system, the primary communications medium is likely to be the Joint Tactical Information Distribution System (JTIDS). In addition to communications, this system will also supply navigation data. The integrated avionics system must be planned to support JTIDS, including all the improvements and modifications likely to occur as the system evolves.

In addition to JTIDS, the generic avionics system must be sufficiently flexible to support various mission-specific communications elements. These would include such things as ASW and airborne early warning (AEW) data links, and any specialized weapon-delivery communications that may evolve as new ordinance systems are devised.

Various radio links to support navigation and guidance of the aircraft must also be supported. Most prominent among these is the NAVSTAR Global Positioning System (GPS). In addition, the avionics system must have the potential to support path-guidance systems such as TACAN, automatic-landing-system communications, and provide the communications capability to operate in controlled airspace.

As is the case for the other functions to be supported by the avionics system, communications elements must be available on a highly flexible and modular basis in order to support functions at necessary levels of survivability and reliability, without excessive replication of elements. These elements must be linked by an appropriate internal data-communications system so that resources can be managed efficiently in hierarchical fashion. In addition, the system must provide the flexibility to support various mission-specific communications systems, and provide the flexibility to allow growth and change as new developments occur.

## 2.2 General Goals and Procedures

The preceding sections have outlined a number of goals specifically related to the various functions that must be supported by a generic fault-tolerant core avionics system. In addition to these goals, there are a number of general attributes that the system must possess.

The system must consistently achieve a high level of fault- and damage-tolerance for each of the functions it serves. However, at the same time, fault-tolerance levels must be no greater than is actually necessary to support each individual function. The design approach must define the required levels of performance, reliability, and survivability for each function to be served. Available system elements, such as sensors, actuators, microprocessors, displays, etc., must be surveyed in order to identify those that can supply the necessary performance. Regions of overlap or intersection between functions, which can be simultaneously served by common elements, must be identified. The use of inertial-navigation sensors to augment flight control is one such example. The types of elements and their levels of redundancy can then be chosen to support each of the functions at the required levels of reliability and survivability. By appropriate choices, the regions of overlap can be maximized, thereby minimizing the numbers of elements, system complexity, and life-cycle costs. This is the very essence of system integration, and it must be the cornerstone of the avionics design.

It is important in this process to apply a rational and systematic approach to fault-detection and identification methods and procedures. Only in this manner can the maximum utilization of all



system elements be achieved. In particular, the choice of failure units is crucial. For purposes of this discussion, a failure unit is the collection of components that is defined to fail as a unit, and which (when one of its components fails) is identified as failed by the onboard redundancy-management system. A failure unit may be very large, such as an entire gimbaled inertial navigator that fails as a result of single transistor failure. By the same token, a single transistor is an example of an extremely small failure unit. The choice of failure-unit size is a tradeoff. There is no totally systematic approach to the definition of appropriately sized failure units; however, a number of factors including unit costs, packaging, repairability, logistics, and systematic redundancy management must all enter into the choices.

The architecture of the integrated system must embody a data-processing and internal-communications configuration that is extremely reliable, survivable, and flexible. Information must be able to be routed (as necessary) to any element of the system. At the same time, the system must establish a hierarchical organization, delegating tasks and responsibility (as appropriate) to the functions served. The information-processing system must implement the fault-detection and isolation algorithms necessary to manage its own redundancy and the redundancy of the sensors, actuators, displays, controls, etc., with which it communicates. It must respond to crew commands, reconfiguring its resources to changing requirements, and must also automatically reconfigure in response to faults and damage.

The integrated avionics architecture will, by its very nature, have lower acquisition costs than a system designed to attain the same performance in a more conventional fashion. However, it is the total life-cycle cost that should be the driver in defining a system. In addition to acquisition costs, the costs of maintenance and logistics are significant in this regard. By defining appropriately sized failure units, and making these units an integral part of the maintenance and logistics procedures, significant savings in these crucial areas can accrue. Since the onboard fault-isolation system identifies the failure unit, the task of diagnosing faults by maintenance personnel can be greatly simplified, with commensurate savings in maintenance costs. Furthermore, the fault isolation is carried out in the actual operational situation, rather than in a simulated test situation back at the base, and the validity of fault isolation is improved, thus

decreasing the incidence of false removals. Additional savings can accrue if the number of unique system elements is minimized. By providing pools of identical elements within the system, the required levels of performance, reliability, and survivability can be attained by a population of units which fall into a very few different classes. The resulting logistics problems are then greatly simplified with simultaneous savings in logistics costs.

Also of significance here is the prospect that a very high level of competition in the procurement of both the system and its spares can be imposed with this approach. By appropriately standardizing interfaces, the modular system elements can be procured in large lots from a number of industrial suppliers. The resulting competitive-market situation will have a significant impact on procurement costs.

The design must have inherent flexibility for growth and change. It must be truly generic in the sense that it can provide the core-avionics functions for a broad range of tactical aircraft from small V/STOL fighters to large CTOL transports. It must be readily adaptable to the various mission configurations implied by this range of vehicles. Finally, it must be capable of adapting to support new functions and subsystems as they are developed, without requiring extensive redesign and modification.



## SECTION 3

### INFORMATION-PROCESSING ARCHITECTURE

This section defines, from a general perspective, the evolving information-processing problem for avionics systems of the 1990s. Technology trends and the opportunities they present for future avionics are examined and criteria to be met by future avionics systems are set forth. Trends in current and future avionics architectures are presented by way of example, and the characteristics of these architectures are compared in terms of the criteria. Finally, a brief description of the proposed baseline for a generic fault-tolerant core avionics system is presented.

#### 3.1 The Changing Information-Processing Problem

The choice of an information-processing system architecture must be made in the context of the information-processing problem. As the problem changes, information-processing system architectures must also change. The purpose here is to characterize this problem for advanced systems in the 1990s time-frame, in order to better understand the ways in which system requirements are likely to evolve. This understanding, together with an appreciation of technology trends and the opportunities they afford for architectural innovation, is a necessary first step in defining an appropriate architecture for the intended application.

Looking ahead to the 1990s, sharply increased demands for avionics systems' digital information-processing resources are foreseen. In part, these increased demands can be attributed to the shift, already well underway, from analog to digital realizations of avionics system signal- and data-processing functions. In part, they can be attributed to a natural growth in the requirements imposed on avionics systems by an increasingly complex military environment. And, in part, they can be attributed to the exploitation of new opportunities afforded by the extraordinary rate of change, which

continues to characterize digital electronics technology, particularly that portion driven by the commercial sector.

As the demands for information-processing services increase, the trend toward decentralization of the information-processing resources is likely to accelerate. Decentralization is one approach to making an increasingly complex computational problem tractable. In addition, decentralization is the inevitable outcome of current technology trends, which make it possible to embed small powerful information-processing devices in a wide variety of heretofore relatively "unintelligent" avionics-system elements. Finally, decentralization is made necessary to ensure survivability in a combat environment.

As the role of the computational elements of the avionics system expands, the need for improved reliability becomes more urgent. The automation of critical avionics-system functions, such as flight control and autoland, requires a highly reliable information-processing system. This requirement for improved reliability is given further impetus by two other trends: the automation of "routine" flight operations to relieve overburdened crews, and the use of computers to make time-critical decisions as weapon sophistication evolves. In addition, the need for improved reliability grows out of the broader requirement to achieve higher levels of availability, and to keep the costs associated with deploying and maintaining military aircraft in the field within acceptable bounds.

As the magnitude of the information-processing problem increases, as the trend toward decentralization evolves, and as the requirement for improved reliability becomes more urgent, the resource-management function of the information-processing system will grow in importance. The responsibility for monitoring the health of the system, containing faults, and allocating resources will have to reside in an absolutely dependable system manager.

There are other ways, as well, to characterize the evolving information-processing problem. The architectures of the future must exhibit greater flexibility in accommodating growth and change. An appropriate balance must be struck between the benefits of standardization, and those of diversity, as a necessary stimulus to innovation.



And the need to make the software development, verification, validation, and maintenance process more manageable must be given special attention.

### 3.2 Technology Trends and Opportunities

An appreciation of current trends in digital electronics technology is required in order to define an information-processing system architecture which is well matched to the information-processing problem in the 1990s. These trends serve to broaden the spectrum of choices available to the avionics-system designer, and to afford opportunities for architectural innovations that may not have been realizable with previously available technology.

The extraordinary rate at which digital electronics technology is evolving makes long-range predictions uncertain. Certain basic trends, however, are apparent. Higher levels of integration are being achieved. As a result, digital electronics' weight, volume, and power requirements are decreasing. Speed is increasing; device reliability is increasing; and efficient, economical hybrid packaging techniques are emerging. Economies of scale are being achieved through the exploitation of large commercial markets. Costs per unit of data-processing power continue to improve.

An opportunity of paramount significance grows out of the current trends toward small, powerful, inexpensive digital computational elements. For the first time, it is possible to envision a system which incorporates an abundance of information-processing resources. This abundance is particularly significant because of the relief it implies in implementing the data-processing functions. The complexity and cost of software development and verification increases significantly as the demand for data-processing resources approaches (or alas, exceeds) the capacity of the data-processing system. Software development and verification costs are decreased significantly when the availability of information-processing resources exceeds the demands for such resources by healthy margins.

The availability of small, powerful, inexpensive digital computational elements also provides a stimulus to decentralization. The imaginative use of embedded processing contributes to a natural partitioning of the complex information-processing function into numbers of relatively simple subfunctions. These subfunctions can be mechanized inexpensively and organized hierarchically, to enforce the degree



of autonomy desired, and to achieve the degree of coordination necessary. The autonomy of these subfunctions, provided locally to sensors, actuators, displays, etc., provides leverage in reducing data-transmission requirements. By confining strictly local tasks to embedded processors, a large overhead burden is lifted from the avionics-systems internal data-communications facility, resulting a significant reduction in bandwidth requirements.

Finally, current trends toward small, powerful, inexpensive, computational elements provide the opportunity to size modules appropriately for fault detection, identification, line replacement, and repair. Declining costs for digital electronics, contrasted with high costs associated with manual fault diagnosis, identification, and repair, suggest a potential benefit if modules can be sized so that fault detection and identification can be reliably automated and the failed module can be economically discarded. In the past, it has been difficult to strike an acceptable compromise between the cost of the module which is discarded, and the costs associated with detecting and identifying failures at that module level. Current trends in digital electronics technology, however, indicate that such a compromise may soon be possible. And even in the case where a module cannot be thrown away economically, a properly sized unit, having its own fault-identification capability, will greatly reduce the cost of maintenance and facilitate module repair.

Electronics reliability is improving on a per-device basis, largely as a consequence of integrating more devices on a single chip. Whereas this might be taken as an indication that system reliability is becoming less of a problem; in fact, there are two important factors which discount that view. The first is an ever increasing functional complexity; and the second is an ever increasing critical dependence on these functions. The primary means of achieving the required reliability for the information-processing function, therefore, will continue to be through the use of redundancy. Current trends bear on how this redundancy can be designed into the system economically. The reliability requirement varies with the criticality of the function to be served (navigation, flight control, pilot display, etc.). Since the requirement is not uniform, allowing the most critical tasks to drive the reliability requirement throughout the information-processing system is unnecessarily wasteful of resources. To date, a compromise has been struck by segregating functions according to their inherent

requirements for reliability and then mechanizing dual, triplex, or quadruplex systems to satisfy the reliability requirement. As the information-processing problem evolves, the inflexibility of this approach and the relatively poor performance-to-complexity ratios (which characterize n-plex structures) will create an impetus to adopt architectures better matched to the changing nature of the information-processing problem. Current trends in digital electronics technology make it possible to pool and dynamically allocate information-processing resources to provide computational services throughout the avionics system at the level of reliability each particular processing task or function requires.

### 3.3 Criteria for Assessing Information-Processing Architectures

In order to assess information-processing architectures, a set of performance, economy, and safety-related criteria are proposed (see Table 3-1). The desirability of designing information-processing architectures which satisfy these criteria is largely self-evident.

Table 3-1. Information-processing architecture assessment criteria.

#### A. Performance-Related Criteria

1. Performance indexes such as throughput, memory capacity, I/O bandwidth, etc.
2. Automation potential
3. Flexibility of utilization
4. Availability

#### B. Economy-Related Criteria

1. Modularity/few unique elements
2. Low complexity, weight, and volume
3. Maintainability
4. Diagnosability
5. Programmability
6. Producibility
7. Low design risk

#### C. Safety-Related Criteria

1. Multiple-fault tolerance/coverage
2. Damage tolerance
3. Low-malfunction correlation
4. Low-fault latency
5. Intermittent fault identification



Prevailing technology and a reluctance to fully integrate resources, however, have heretofore posed serious obstacles to achieving architectures which adequately measure up to these criteria. Given present trends in digital electronics technology and the opportunities these trends afford, more satisfactory performance can be realized. The following paragraphs describe representative classes of architectures, and evaluate each in terms of the criteria of Table 3-1. That discussion is followed by a description of a generic, hierarchical, fault- and damage-tolerant architecture which is well matched to the scope and requirements of the information-processing problem envisioned for avionics systems in the 1990s.

### 3.4 Avionics System Computation Architectures

At the present time avionics system architectures are rapidly evolving. This is in large measure due to the significant advances in electronics technology that have occurred in the recent past, and which are continuing. Simultaneously, there is a synergism between capabilities and requirements so that as capabilities increase requirements grow apace. The dynamic interaction between the two has spurred a number of significant avionics architecture developments. This section presents brief descriptions of the various avionics system redundant architectures which have been used in the past, or which are currently deployed on operational or prototype vehicles, or which are in the initial stages of development.

#### 3.4.1 Dissimilar Primary and Secondary Resources

The use of separate independent primary and secondary resources has been a long-standing well accepted approach to avionics system redundancy. Manual backup of an automatic system, such as an autopilot, is the most familiar example of this approach. The human operator is responsible for both diagnosis of a failure, by recognizing inappropriate behavior, and reconfiguration, which is accomplished by the simple act of disengaging the system and performing the flight-control function manually.

The guidance and navigation system of the Apollo Lunar Landing Module was an example of an automated dissimilar backup system. The primary system consisted of a computer, inertial platform, optics, a display-keyboard crew interface, and rendezvous and landing radar units. Manual guidance could be supplied by the crew via a set of

hand controllers. Attitude control was provided by a set of reaction control thrusters. Rocket engines provided the lunar descent and ascent thrust, and both they and the reaction thrusters were used for control during thrust phases.

Redundancy was provided during lunar descent and ascent by an abort guidance system consisting of a computer, inertial package, and display-keyboard, all different from those of the primary system. The hand controllers, radar, engines, and reaction jets were shared with the primary.

With the exception of a data link from prime to backup, which allowed fine tuning of the backup state vector, etc., the two systems interfaced only at points of common hardware. No data could be passed from system to system at these points.

Control was handed from the prime to the backup system via explicit crew action. The crew and ground personnel were solely responsible for manually monitoring the health of both systems and making the switching decision. The capabilities of the two systems were nearly comparable, but the design, production, procurement, validation, training, maintenance, and so forth, were all essentially distinct, and required duplicated effort and cost.

A more current example of this approach is the digital flight-control system of the Saab Viggen fighter aircraft. A single primary digital flight-control computer provides a broad range of flight-control modes and capabilities during normal periods of operation. The health of this computer is constantly checked by a simple, but highly reliable, monitoring system, which requires the flight computer to perform a separate on-line feedback control function for a simulated system. The performance of this control function is monitored, and if specific criteria are not satisfied the monitor switches over to a backup system.

An important emerging requirement, illustrated by way of contrast between these systems, is the need for rapid, automatic fault identification and system reconfiguration in modern high-performance aircraft. In many failure situations, a significant body of information from many sources must be analyzed in order to diagnose a fault. Often the crew is incapable of assimilating the data and making a decision quickly enough to avoid disaster. Thus automatic diagnosis is used in the Viggen, whereas manual procedures were sufficient in Apollo.



#### 3.4.2 Replication of the Primary Resource

Many current and projected avionics systems utilize this category of redundant system architecture. The dual approach of identical instruments for pilot and copilot and redundant communications and radio/navigation transceivers is a prime example in current aircraft.

The NASA experimental F-8C research aircraft is a more advanced example. This vehicle incorporates a triply redundant digital fly-by-wire flight-control system, as shown in Figure 3-1. Each of the three strings contains a set of inertial sensors, a digital computer and a signal interface unit (IFU). Equipment failure is detected by voting across the three strings.

The strings are cross-strapped in the following ways:

- (1) Each IFU receives inputs from all three sets of inertial instruments.
- (2) The computers can pass information directly among themselves.
- (3) The control surface actuators have multiple input ports and a mid-value selector associated with each input port. Each mid-value selector receives the control commands from all three strings.

Going to a triplex voting system provides almost 100-percent coverage of the first failure. The cross-strapping at different points allows the system to tolerate certain sets of multiple failures. A dissimilar analog backup flight-control system is used in the event of two computer failures.

A similar digital flight-control research program was pursued by the Air Force, utilizing an A-7D aircraft. This system contains dual redundant digital computers with the conventional analog A-7D flight-control system as backup. First failure of a computer is detected by comparison of the outputs of the two computers. Fault isolation and reconfiguration after the first failure and fault detection of a second failure are both performed by an extensive built-in test system.

Another example of primary resource replication is the prototype F-18 aircraft, which has a quadruply redundant digital flight-control configuration operating in dual-dual fashion. The four computer channels operate as two pairs, with the outputs of each pair compared to detect faults. In the event of two computer failures, the system reverts to an analog backup system.

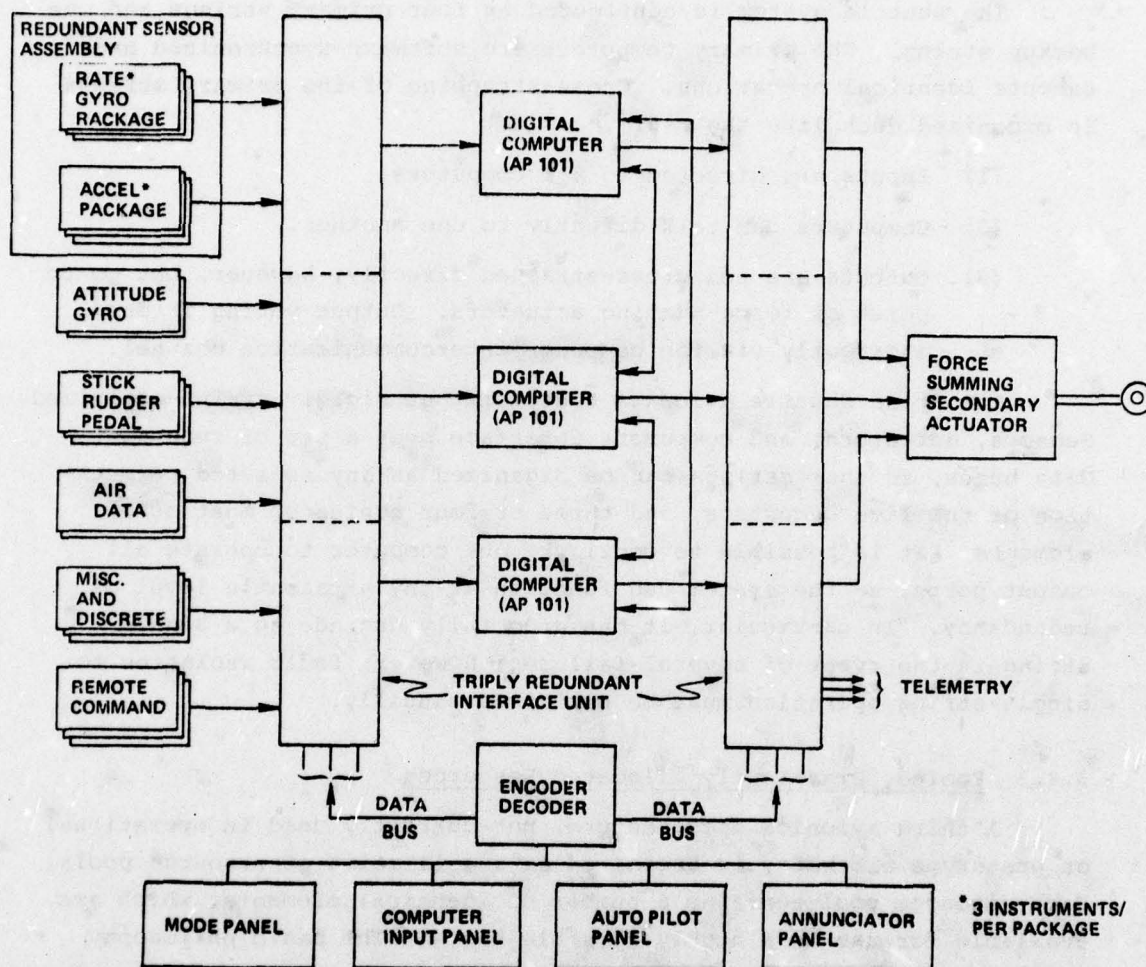


Figure 3-1. F-8 digital fly-by-wire flight-control system.



The aforementioned examples were of flight-control systems, and hence represented only subsets of a total avionics system. The NASA Space Shuttle contains a total core avionics system in the sense that it includes navigation, communication, and display elements as well as flight control.

The shuttle system is configured as four primary strings and one backup string. The primary computers are software-synchronized and execute identical operations. Cross-strapping of the primary strings is organized much like the F-8.

- (1) Inputs are directed to all computers.
- (2) Computers can talk directly to one another.
- (3) Outputs are not cross-strapped directly, however, but go to ports of force-summing actuators. Output voting is done indirectly via the computer intercommunication channel.

The Space Shuttle avionics system is not rigidly string-organized. Sensors, actuators, and computers interface over a set of redundant data buses, so that strings can be organized as any selected permutation of the five computers, and three or four copies of most other elements. It is possible to configure one computer to operate all output ports, so the system can function at any attainable level of redundancy. In particular, it can gracefully degrade to a single string in the event of several failures; however, fault isolation to single-string operation must be performed manually.

#### 3.4.3 Pooled, Dynamically Allocated Resources

A third avionics architecture, not currently used in operational or prototype aircraft, is organized as a collection of resource pools. Each resource pool contains a number of identical elements, which are available for use on a highly flexible basis. The basic philosophy behind this approach is that no distinction is drawn between elements in terms of dedication to specific functions. Rather any element can serve any function for which it can provide useful capability.

The pooled elements are the line-replaceable units within the avionics system. All internal fault isolation is carried to the level of these units, and maintenance and logistics procedures are designed around these as the basic system elements. In addition, the pooled units are basic building blocks which are interconnected (in hierarchical fashion) to perform the necessary avionics-system functions.

The interconnections between units are not static, but can be altered by the system in real time to respond to changing requirements and loss of capabilities due to failures and damage. Embodied within the design is a comprehensive redundancy-management function which identifies faults to the level of the line-replacement units, and reconfigures the system to isolate failed units.

Each resource pool of units is represented at a level of redundancy reflecting the possible functions that can be served by the elements of that pool. In many instances, elements can serve multiple functions. The prime example here is a small computer or microprocessor, which can serve any function for which it is programmed. By loading appropriate code in real time, and providing the necessary input/output interfaces, the computer can serve any function for which its capabilities are sufficient. Thus, for example, a processor is not dedicated to flight control, but can be directed to serve navigation, display, or communications functions as needed. Similarly, a rate gyro may serve both as an inertial-navigation element and as a flight-control element. Numerous other examples of multiple roles for radio transceivers, displays, controls, etc., can be identified.

A particularly significant aspect of the pooled resource approach is the determination of the size and capability of the units that constitute the resource pools. The determination is, in effect, a comprehensive tradeoff between many factors. The size and complexity of the unit must be neither too large nor too small. Too large a unit means an excessive loss of capability as a result of a single-point failure within the unit. A large, single, highly capable computer, or a gimballed inertial navigator, are examples of inappropriately large units. By the same token, too small a unit can impose an excessive burden on fault-isolation mechanisms, to identify the fault to that unit level. A transistor or a gyro-spin motor are examples of inappropriately small units. In addition to these, a number of other factors, such as availability, logistics, performance requirements, packaging, and ease of maintenance must be taken into account in determining unit size. All these factors ultimately impact life-cycle costs.

### 3.5 Comparison of Avionics Architectures

The preceding sections presented three primary approaches to avionics system design. In this section, a series of comparisons



will be drawn between the three architectures in terms of the criteria listed in Table 3-1. The criteria are broken down into categories of performance, economy, and safety. These categories are not mutually exclusive in that some of the criteria can comfortably fit under two or more categories and certain criteria are mutually supportive.

Although all three architectures have advantages and disadvantages, the "Pooled, Dynamically Allocated Resource Approach" provides a very high level of flexibility to adapt to changing requirements. This, plus the potential of this architecture to significantly benefit from the emerging electronics technology, place it in a leading position. The following subsections attempt to point out how and why this approach has the potential to significantly exceed the other two architectures in terms of these important criteria.

To facilitate the discussion, the terms designating the three architectures will be shortened. The "dissimilar primary and secondary resource" architecture will be called "dissimilar", the "replication of the primary resource" architecture will be called "repetitive", and the "pooled, dynamically allocated resource" architecture will be called "pooled".

#### 3.5.1 Performance-Related Criteria

These criteria relate to the levels of performance required of each of the functions served by the avionics system.

##### 3.5.1.1 Normal Performance Indices

This is a broad category containing the numerous performance criteria such as throughput, memory capacity, bandwidth, etc., that the avionics system must possess in order to perform all of its functions. They are not architecture-sensitive criteria, and any of the aforementioned architectures, if provided sufficient capabilities, can satisfy these requirements. This set of criteria is only brought forward to make the point that the three architectures are equally capable of satisfying these requirements.

##### 3.5.1.2 Automation Potential

Automation potential is the ability of an architecture to provide an appropriate level of reliability and survivability to permit the automation of critical functions. The "dissimilar" architecture with

single-failure fault tolerance is inadequate here. The repetitive and the pooled architectures, possibly incorporating dissimilar back-ups, can both be configured with sufficient redundancy to satisfy this criterion. However, because of its ability to assign elements freely to perform necessary functions, the pooled architecture has the potential to satisfy this criterion more economically and with less complexity than the repetitive architecture.

#### 3.5.1.3 Flexibility of Utilization

The pooled architecture is far more effective in this case than the other two approaches.

There are two types of flexibility that can be addressed. The first is real-time flexibility to adapt to a changing situation during a mission or in flight. The concept of pools of units provides in-depth flexibility. Since no unit is specifically identified with a particular task or function, all units are essentially available to support any required function, and the maximum flexibility is afforded within the constraints of numbers of available units.

The second type of flexibility is the ease with which the system can be modified for growth and change. With the pooled approach, modification means changing the constituents of pools, sizes of pools or altering by adding or subtracting pools. To the extent that internal-communications-system flexibility permits, this can be a very straightforward process. A rigidly string-organized system (i.e., triplex- or quad-redundant) can be amenable to changing particular types of elements, but is not economically amenable to adding additional redundancy of particular elements. A non-string organized "repetitive" system can have change flexibility equivalent to a pooled system. It is important to realize that this type of configuration begins to approach the pooled architecture.

#### 3.5.1.4 Availability

Availability is a measure of the ability of a system to be used when needed. Factors impacting availability are inherent reliability, maintainability, and ease of logistics support.

Because of its ability to flexibly configure its resources, the pooled system has the potential to provide the highest level of availability. Since no element is specifically assigned to a function, all



elements can serve any function to which their capabilities are useful. Thus, for a specified number of faults within the system, the pooled architecture, with elements serving multiple roles and flexibility to reconfigure around faults, has the greatest chance to complete a mission successfully.

In addition, its ability to identify faults to the level of line-replaceable units makes the pooled architecture easy to maintain, and appropriately sized line-replaceable units ease logistics.

### 3.5.2 Economy-Related Criteria

These criteria most directly affect system life-cycle costs.

#### 3.5.2.1 Modularity

It is most desirable, in terms of procurement, maintenance, and logistics, that the avionics system be configured from as small a set of unique modules as possible. The very nature of the pooled architecture lends itself readily to minimizing the number of different pools, and hence the number of unique modules. Pools can serve multiple tasks and, by appropriate choice of elements in a pool, the array of functions served can be maximized, and the number of pools minimized. Although the other architectures can also lend themselves to minimizing unique modules, the greatest advantage in this area is afforded by the pooled architecture.

#### 3.5.2.2 Low-Complexity, Weight, and Volume

The total set of functions served by the avionics system overlap in the sense that more than one function can be served by a single unit within the system. Often computers, inertial sensors, radio transceivers, etc., can serve both mission and flight functions. By purposely choosing units to serve multiple roles, both performance and reliability can be achieved without excessive complexity, and its associated weight and volume penalties. As faults occur, the system can be reconfigured, gracefully dispensing with less critical mission functions so that remaining resources can support flight-critical tasks. The pooled architecture, with its dynamic reconfiguration capability and pooled resources, provides the flexibility to make maximum use of all elements, and hence has the greatest potential to minimize complexity, weight, and volume.

#### 3.5.2.3 Maintainability

The fault-isolation procedures of the pooled architecture are specifically designed to identify failures on-line, to the level of line-replaceable units. The fault-isolation process occurs in real time with the aircraft in its operational environment, and hence a high level of validity of fault isolation can be achieved, as compared to the usual maintenance procedure involving after-the-fact testing and checkout.

Line-replaceable units (LRU) are sized to facilitate ease of maintenance, and the entire maintenance and the logistics planning and operation are based on the pooled units as basic system elements. This, plus the fact that the number of different LRUs is minimized, greatly simplifies the overall maintenance problem. The dissimilar and repetitive architectures can also lend themselves to ease of maintenance, but not to the degree possible with the pooled architecture.

#### 3.5.2.4 Diagnosability

This aspect or criterion for the avionics system reflects the ease with which faults can be identified for maintenance purposes. It is more a design aspect than it is an inherent property of particular architectures. In the past, fault isolation and built-in tests were often an afterthought appended to the design. However, in future systems, they must be an integral part of the design process from the outset. All three architectures lend themselves to fault diagnosis; however, the pooled design specifically incorporates this factor as a significant part of the entire design process.

#### 3.5.2.5 Programmability

Architectures can have a significant impact on the orderly development of software, its verification and validation. The most important aspect of a software development effort is the systematic partitioning or modularization of the job. A modular architecture imposes a natural partitioning on the software that greatly enhances this process. Since the pooled approach is by its very nature the most modular of the architectures, it has the greatest potential for advance in this area.



#### 3.5.2.6 Producibility

Relative to the Navy's desire for long-term economical acquisition of avionics systems for a large fleet of aircraft, producibility is largely dependent on the ability to establish highly competitive procurement of the various elements of the avionics system. The pooled architecture, consisting of relatively few pools, containing rather large numbers of identical elements, can provide the basis for establishing this competition. Large numbers of units will characterize each buy, attracting many potential suppliers. Fewer different types of units will be required, resulting in a reduced inventory of spares. The Navy can, thus, take maximum advantage of the innovative cost-reduction methods that are inevitably stimulated by competition. The Navy owns the architecture, and will be able to supply the parts for it on a piecewise competitive basis.

#### 3.5.2.7 Design Risk

Of the three architectures described, the pooled approach is the newest, and hence has the smallest base of experience. Both the dissimilar and repetitive approaches have been used in prototype and operational aircraft. While extensive analyses, simulations, and prototype experiments have demonstrated the potential and feasibility of the pooled approach, it has not been brought to the flight-test stage of development. A comprehensive flight-test demonstration is an essential element in reducing the level of risk that currently exists.

#### 3.5.3 Safety-Related Criteria

These criteria address the critical issues of fault and damage tolerance.

##### 3.5.3.1 Multiple Fault Tolerance

Both the repetitive and the pooled architectures can be provided with sufficient levels of redundancy to tolerate multiple faults. Similarly, both architectures can embody redundancy management procedures to identify failures and reconfigure the system to isolate faulty elements. However, the pooled architecture, (with no dedication of units to specific functions and flexibility to allocate resources on a priority basis) can provide a higher level of fault tolerance at a given level of complexity; or reduced complexity for a given level of fault tolerance.

#### 3.5.3.2 Damage Tolerance

The most effective means of attaining tolerance to localized battle damage is through separation of avionics system elements. This requirement, if applied literally to a repetitive architecture, would result in a considerable overhead penalty in terms of numbers of elements. Since the pooled architecture generally contains smaller units, as compared to the other architectures, greater freedom in location of elements within the aircraft is afforded by this approach.

#### 3.5.3.3 Low Malfunction Correlation

Malfunction correlation poses the greatest threat to all fault-tolerant architectures. The very basis of fault tolerance is the assumption that the system can be designed so that failures are independent events. A correlated failure that affects all redundant copies of a particular type of unit immediately thwarts the purpose of redundancy.

The dissimilar architecture is, by its very nature, highly immune to correlated malfunctions. The repetitive and pooled architectures must be designed with special care to eliminate all correlated malfunction mechanisms.

#### 3.5.3.4 Low Fault Latency

Fault latency manifests itself in a fashion that is similar to a correlated malfunction. A latent fault might occur, for example, when a unit is designated as a spare, and is not exercised for a period of time. The unit may fail during that period, and the fault could go undetected. The latent fault poses a special threat because when the spare is brought on-line to replace a detected fault, the spare malfunctions.

The dissimilar architecture typically operates with its backup system inoperative for long periods of time. Latent faults can accumulate and defeat the redundant strategy when a failure occurs in the prime system. The repetitive system tends to use its redundant units in parallel fashion, all performing identical tasks. If the particular task does not exercise a certain facet of these units, so that a failure is not observed, latent failures may accumulate. The pooled architecture routinely reconfigures itself to uncover latent faults, and hence it has the best chance of purging them.



#### 3.5.3.5 Intermittent Fault Identification

Intermittent faults are the most difficult to diagnose. Rapid detection and diagnosis, and special demerit procedures, or other record keeping methods, must be used to identify faults in this class. In some sense, the intermittent fault is a type of latent fault, and (for the same reasons as those already mentioned) the pooled architecture has the best potential for effectively handling these failures.

#### 3.5.4 Summary of Architecture Comparisons

The pooled architecture has some significant advantages over the dissimilar and repetitive configurations. While there seems little distinction between the three forms in terms of the normal performance indices (i.e., throughput, memory, bandwidth, etc.), the pooled architecture can provide automation potential with less complexity, and it is far superior in terms of flexibility of utilization. This same flexibility provides advantages to the pooled architecture in terms of availability, reliability, and survivability.

Modularity of the pooled architecture is superior; it has the potential for lowest complexity, weight, and volume, and its design is directly responsive to provide ease of maintenance through fault diagnosis. Modularity of the pooled architecture also imposes a natural modularization on the software, which is the most important facet of efficient software development. The reduced number of unique elements in the pooled system provides the basis for a highly efficient procurement program. Together these characteristics imply the smallest potential life-cycle cost for the pooled architecture.

In terms of the safety-related criteria of multiple fault tolerance and damage tolerance, the pooled approach has significant advantages over the others. While the dissimilar system is more naturally resistant to correlated malfunctions, recent analyses, simulations, and tests have indicated the potential of the pooled architecture to be designed to resist this class of faults. The pooled approach is better suited to latent and intermittent fault identification.

As is so often the case, the path of greatest potential gain also represents the largest risk. Although it has received considerable attention in terms of analyses, simulations, and experimental prototypes, the pooled approach has not been flight tested, and has the smallest base of experience. However, results to date indicate

that this avionics architecture is practical and has the potential to satisfy the emerging requirements of the 1990s, while taking maximum advantage of the technology advances that are likely to occur in that time frame.

### 3.6 Information System Architectural Baseline

This section presents a brief description of a fault-tolerant avionics architecture which is based on the pooled approach. The baseline is presented by separate discussions of the major information system components: fault-tolerant computers, embedded computers, and the data-transmission network. Detailed discussions of the elements of the baseline are deferred to Appendix 3-A and Section 7.

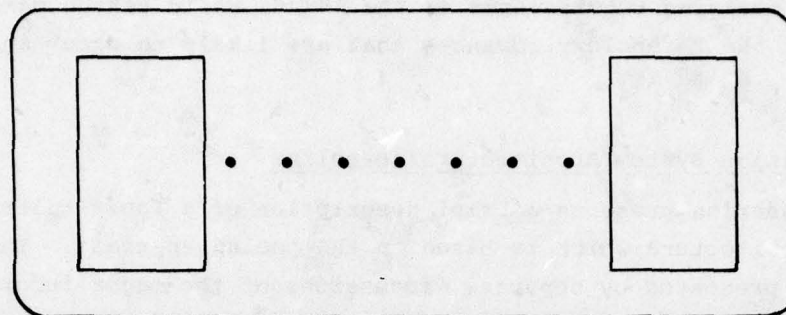
#### 3.6.1 Fault-Tolerant Computer Complexes

The management of the system, as well as the flight-critical computation for the core avionics system, is to be performed by two identical, physically separated, fault-tolerant computer complexes. The principal motivation for having two complexes is for damage tolerance, but some additional protection is also afforded for spontaneous faults. Each of the fault-tolerant computers is a multicomputer or multiprocessor, organized for very high dependability of continuous computation with transparent means of detecting and identifying faulty modules, and of reconfiguration and recovery.

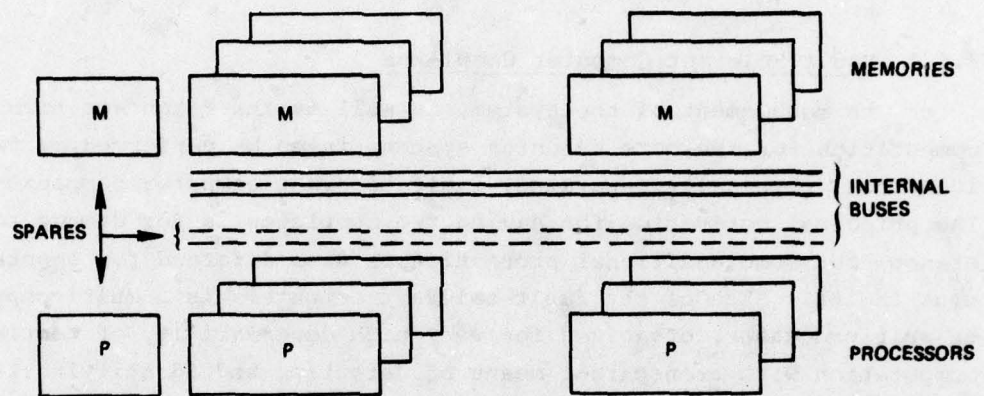
The fault-tolerant multiprocessor (FTMP) architecture can be thought of as an umbrella for a pool of computers to operate with shared resources for detection, identification, and recovery. The conceptual form of the FTMP is shown in Figure 3-2, in which (a) depicts the single LRU type concept, i.e., each FTMP consists of a number (e.g. 12) of identical LRUs, plus a passive backplane. Then, (b) illustrates the processors and memory modules in these LRUs, arranged in triads with a redundant internal bus and on-line spares, while (c) shows I/O access units (also contained in LRUs) located along the internal bus.

The net logical effect of this organization is shown in Figure 3-3, which shows three logical processors (actually processor triads), each of which is shown taking responsibility for one of the network ports. Spares and other special features for fault tolerance are not indicated, but it is implied that these triads are continually tested

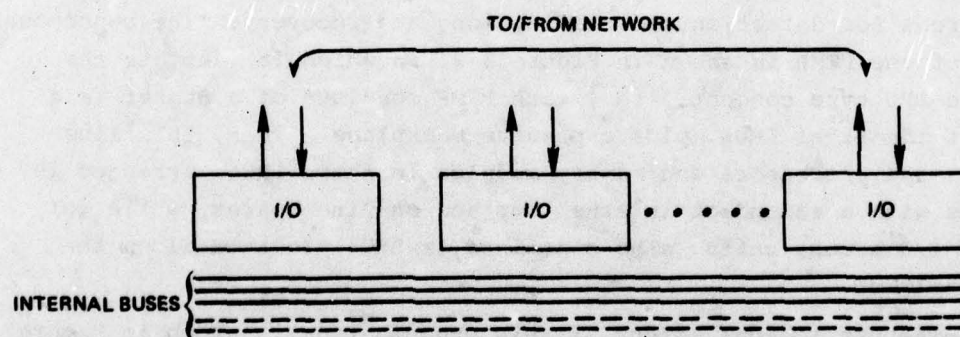




(a) IDENTICAL INDIVIDUAL LRU's



(b) LOGICAL ORGANIZATION OF TRIADS



(c) I/O ACCESS

Figure 3-2. Fault-tolerant multiprocessor concept.

and reconfigured to working status by processes that are essentially not visible to the applications programmer. Thus, the three triads are virtual computers of very high reliability, which communicate via a common global memory of equally high reliability. This arrangement is capable of resembling three independent computers, each of which manages a separate functional partition of the system, such as inertial navigation, flight control, communication-NAVITATION, resource management, display, and so forth. In this way, the multiprocessing nature of the FTMP is de-emphasized, in favor of its multicomputer character. Each processor will possess ample local memory to minimize common memory traffic.

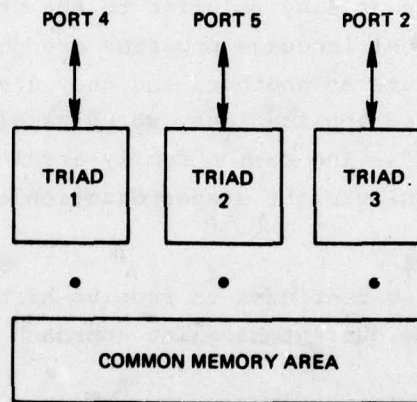


Figure 3-3. FTMP with quasi-dedicated processor triads (redundancy not shown).

The previous discussion provided a broad general description of the FTMP. In order to make it a viable element of a fault-tolerant avionics system, it must satisfy the criteria set forth in Section 3.3. To a great extent, its ability to satisfy these criteria depends crucially on its design details. Appendix 3-A presents a concise description of the FTMP design, along with analyses of its reliability and a documentation of analyses, simulations, and tests in support of its development.

### 3.6.2 Embedded Computers

The baseline system assumes the existence of a computer in every independent sensing and effecting component. In most cases, such as



gyros or actuators a microprocessor is sufficient. A few, such as JTIDS, require high-speed signal and data processing. Some, such as displays and engines, require substantial data processing. It is clear that no single standard embedded computer will properly fit all needs. It is less clear, however, whether or not a standard family will be appropriate.

The problem with embedded computers is that it is not easy to create and enforce a standard architecture, yet it is imperative to be able to maintain hardware and software over the system lifetime. There are various possible compromises, such as choosing a number of standard architectures, or adopting a common high-level language.

There seems to be no easy solution to the standardization problem, however. High-level language programs are not easily transportable from one architecture to another, and they are apt to be awkward for the typical embedded applications, which usually involve time-sensitive "bit-pushing". The common family architectures (CFA) approach is somewhat better than outright standardization of a single architecture.

While this problem continues to receive high-level attention throughout the DoD, the current baseline approach recognizes the following:

- (1) Standardization of computer architectures, although difficult, is essential to the integrated-system approach because of its use of a single fault-tolerant computer architecture.
- (2) The Navy will possess the tools and resources needed to maintain the FTMP software. The FTMP may employ any desired instruction repertoire subject to certain augmentations.
- (3) For any embedded computers that are to be Navy-maintained, the preferred architecture is always a family sibling of the FTMP, supportable with the identical tools and resources.
- (4) Where necessary, Navy-maintained embedded computers will use other repertoires with other architectures. If at all possible, these shall be chosen from within the CFA family.
- (5) For embedded computers that are not to be Navy-maintained, the considerations of reliability, environmental tolerance, and line-replacement potential will apply, which would apply to any device inside a vendor-designed product.

### 3.6.3 Data Communications

The baseline internal data-communications system is a mesh network using standardized node electronics and point-to-point serial full-duplex links. The links may be electric or fiber-optic, with the former favored for any early flight demonstration and the latter for flight systems of the 1990s.

Each system element (i.e., gyro, transceiver, etc.) is connected via its embedded processor to a node in the network. The network can be likened to a telephone switching system. By providing a high level of connectivity, a multiplicity of possible paths can link the FTMPs to various system elements connected to the nodes. At any given point in time, switches within the network nodes configure a virtual data-communications bus, to allow communication between the FTMPs and all system elements. In response to failures or damage, a new bus can be grown within the network, isolating failed elements and connecting to survivors.

The core system would contain of the order of 50 to 100 nodes, depending on the aircraft, its equipment and functions to be provided. The geometry of the linkage will depend on which of several alternatives are chosen for the particular system. The fundamental baseline is a single regular mesh that will support a single path which could be readily reconfigured. The first level of enhancement of this fundamental baseline is to provide dedicated links to support specific high-bandwidth or fast-reaction requirements. Such links may or may not terminate at the FTMP. Dedicated paths might be backed up by a limited multipath capability. The next level of enhancement of this fundamental baseline is to provide multiple meshes. This would deal with the case where the system is partitioned into a number of distinct (or nearly distinct) regions. A single port from one of the FTMPs would be assigned to communicate with the elements of a single region. The network linkage would either favor these regional divisions topologically, or else the number of links per node would be made large enough to support interlaced paths in a single general mesh. A detailed discussion of the network and its key node elements is presented in Section 7.

### 3.7 Summary

Sharply increased demands on information-processing resources will characterize future avionics systems. At the same time, advances in electronics technology will have a significant impact on volume,



weight, and power requirements, providing the potential to satisfy these requirements. The problem for future avionics systems will be to configure architectures that can attain the required goals at appropriate levels of reliability and survivability, at affordable life-cycle costs.

Classical redundancy structures are not adequate to accomplish the desired dependability with economy. A pooled-resource approach that uses embedded computation, a rigorously redundant data-distribution system, and two fault-tolerant computing resource pools, is proposed herein as the baseline for further development.

This baseline will impact numerous Navy technology programs, most of which are mentioned in connection with material covered in later sections. The impacts in most cases are not great. Of specific relevance to this section is the Navy's Information Handling System program. Major goals of this program are to develop a distributed avionics processing architecture which has high fault tolerance, low software costs, and exploits the advances in large-scale integration (LSI) technology. An important aspect of this effort is the development of a methodology for partitioning the avionics processing tasks. Additional effort is being given to defining standardized microprocessor languages. Also being pursued are a number of efforts to develop simulation and evaluation tools, by which various avionics architectures can be tested and evaluated.

All these efforts are consistent with the pooled architecture baseline previously defined. Of particular relevance is the effort to partition the avionics processing task. The partitioning of tasks is crucial to the pooled approach because tasks must be appropriately sized for the processors constituting the resource pools. Also, the effort to standardize microprocessor software will be extremely important for any Navy-maintained software in embedded processors.

## SECTION 4

### INSTRUMENTATION AND REDUNDANCY MANAGEMENT

#### 4.1 Introduction

The core components of the generic avionics system must support a variety of functions. As discussed in Section 2, the nature of an integrated, highly reliable, survivable system tends to blur the distinction between the functions served by various system elements. In order to determine the configuration alternatives it is necessary to survey the following:

- (1) The requirements of the system, including functional, fault-tolerance, and survivability requirements.
- (2) The equipment that can perform the various functions.
- (3) The various methods of fault detection and isolation (FDI).

The necessity for surveying the system requirements and applicable equipment is clear. Perhaps less clear, but equally important, is the need at the outset to stipulate the fault-tolerance and survivability requirements, together with the relevant forms of FDI. Earlier conventional designs have tended to concentrate on a functional black-box approach to equipment selection, leaving the solution of the problems of fault-tolerance, survivability, and redundancy management to ad hoc techniques. Historically, that approach (although successful in meeting requirements) has led to systems in which many types of components have proliferated, unnecessary replication of components has prevailed, and minimal communication has existed between large aggregates of equipment. It is the goal of the generic avionics system to arrive at a core configuration that satisfies its requirements in an efficient manner that is cost-effective over the life of the aircraft. In fact, cost-effectiveness must be considered a primary criterion for judging the success of the generic avionics approach.



As discussed in the following sections, the areas of functional requirements, fault-tolerance, survivability, and redundancy management are intimately interrelated and heavily impact avionics-system configuration. Because of functional overlap between component types, the generic avionics system employs graded redundancy; i.e., instrument replication is allowed to vary with instrument type and function served. Fault-tolerance criteria, whether derived from statistical analysis or engineering judgment, only stipulate minimum equipment replication requirements; and only through information-efficient high-coverage FDI techniques can this minimum redundancy be realized in the system (i.e., a minimally-redundant system). In addition, some approaches to survivability (in particular, spatial separation of instruments) put severe limitations on the levels of instrument failures that can be isolated using instrument output comparisons. Thus, it is only through the judicious use of both hardware and software FDI that the computational power of the generic avionics system can be utilized to meet the fault-tolerance and survivability requirements with minimum life-cycle cost.

#### 4.2 Requirements of the Core Avionics System Design

There are three general categories of requirements for the core avionics system: functional requirements, fault-tolerance requirements, and survivability requirements. As discussed in Section 2.1, the core of the generic avionics system must support the four basic functions of flight control, navigation, display and control, and communications. This section deals specifically with flight control and navigation. Section 5 addresses display and control, while Section 6 addresses communication.

##### 4.2.1 Flight-Control and Navigation Functional Requirements

The flight-control function must assure a well behaved aircraft, from the point of view of pilot handling, and must also support a number of path-control functions, including fire-control and weapon delivery, landing, and various mission-oriented automatic navigation modes (such as terrain-following and low-altitude flight over water). Since many of the flight-control tasks are flight-critical, the avionics system must support the flight-control function to levels of reliability and survivability commensurate with the reliability and survivability of the airframe itself.

The navigation function must support accurate target location and engagement, guidance to the vicinity of a desired landing site, and position, velocity, and attitude information transfer to aircraft weapons. In most cases, the navigation function is only mission-critical, and its reliability need not be as high as that of the flight-control function.

#### 4.2.2 Fault-Tolerance and Survivability Requirements

There are several requirements relating to fault-tolerance and survivability of the various functions performed by the core avionics system that have a strong impact on overall system architecture. Some of the more significant of these requirements are as follows:

- (1) Number of individual, independent faults that must be tolerated in the absence of battle damage.
- (2) Number of individual, independent faults that must be tolerated in the presence of localized battle damage.
- (3) Performance capability required following battle damage.
- (4) Stand-alone performance required in the absence of external aids.

The requirement of fault tolerance, and the requisite existence of redundant components\*, may arise from at least two rationales—one probabilistic, and the other ad hoc. In the former, the desired probability of function success cannot be met with a single state-of-the-art component. Consequently, a sufficient number of instruments of that type must be onboard in order that the probability of the loss of all components (and resulting loss of function) is lower than the desired level. In the ad hoc approach, functional operation in the face of some number of consecutive single-point failures is felt to be a desirable philosophy, regardless of probabilistic arguments. Whether either or both of the aforementioned rationales is used, it is fair to say that redundancy will exist within the avionics system, particularly to support the flight-critical functions of flight control and automatic landing.

There are two forms of redundancy that can be utilized to meet a requirement for fault-tolerance: direct redundancy, and analytic

---

\*The connection between fault tolerance and redundancy is made in the following discussion.



redundancy. Direct redundancy involves the replication of like-sensor types, while analytic redundancy involves the redundancy arising from functional relationships among the inputs to various sensors. To illustrate analytic redundancy, attitude gyros measure Euler angles indicating the vehicle attitude, while rate gyros measure components of the vehicle angular velocity. These two quantities are related via non-linear differential equations, and thus the outputs of the attitude and attitude rate gyros should be similarly related. Such relationships can be exploited for failure isolation, as discussed in Section 4.4.3, and the use of analytic redundancy for failure isolation allows the utilization (to the extent practical) of all of the information available in the total sensor complement. It is important to note, however, that sensor-failure isolation using analytic redundancy tends to be more complex and slower than isolation using direct redundancy. The advantage of analytic redundancy is that in many applications it allows the use of fewer instruments of each type to meet specified failure-tolerance requirements, with resultant savings in weight, volume, power, and acquisition and maintenance costs.

For the case of direct redundancy using single-degree-of-freedom inertial instruments measuring different components of the same n-dimensional physical vector quantity, it can be shown that if only like-instrument output readings are used: <sup>(4-1)\*</sup>

- (1) At least  $(n + 1)$  instruments are required to detect the hard failure of a single instrument.
- (2) At least  $(n + 2)$  instruments are required to isolate the hard failure of a single instrument.

Thus, for example, strapdown accelerometers measure components of linear acceleration, a three-dimensional vector. It follows that four or more instruments are required to detect the failure of an instrument, and five or more instruments are required to isolate that failure. Note that these figures indicate the minimum number of instruments necessary, and a greater number of instruments may be required if poor geometry (e.g.,

---

\* Superscript numbers (4-1, 4-2, etc.) refer to similar numbers in the List of References at the end of the section.

parallel input axes) is employed. For the case of measurements of a scalar quantity, such as altitude, it follows that two or more instruments are necessary to detect a single instrument failure, and three or more instruments are necessary to isolate the failure.\*

It is difficult to anticipate for the time frame of the 1990s function and performance requirements following battle damage. A good example of the current and near-term philosophy is the Air Force multi-function inertial reference assembly (MIRA).<sup>(4-2)</sup> Although MIRA will perform flight-critical functions, it consists of only a single cluster of instruments, and relies upon any of the following three techniques for survival after limited battle damage.

- (1) Armor protection (approximately 100 pounds).
- (2) Shielding by other equipment.
- (3) Location adjacent to the pilot seat.

A fourth survival technique (cited in Reference 4-2 as feasible, but not currently planned) involves the use of two or more MIRA units separated spatially. A distance of 30 inches is stated as being sufficient for one unit to survive a 23-millimeter shell hit on the aircraft. The use of spatial separation to achieve battle-damage survivability is also suggested in the feasibility study of an integrated control-sensor subsystem for advanced V/STOL aircraft.<sup>(4-3)</sup> In the approach of Reference 4-3, a set of six flight-control sensors of each type (rate gyro and accelerometer) is arranged on a bulkhead in three two-packs, with each two-pack at the corner of an equilateral triangle 35 inches on a side. Such an arrangement allows undegraded flight-control function performance following two successive single-instrument failures, or a single limited-damage event.

#### 4.3 Instruments Available

In this section, the available instrument types that can satisfy the various functional requirements are discussed. Such a discussion is necessary for two reasons. First, most functional requirements do not uniquely determine the instrument types required. Second, knowledge of the candidates for the avionics-system instrument complement is imperative in order to make a final avionics complement selection

---

\* Another approach to instrument-failure isolation involves self-test, which will be discussed in more detail in Section 4.4.1.



that results in fulfilling the functional, fault-tolerance, and survivability requirements in a manner that is cost-effective over the life of the aircraft.

#### 4.3.1 Navigation Instruments

There are two broad classes of equipment that can be used to perform the navigation function: inertial reference assemblies (IRAs) and radio-navigation aids.

In general, an IRA is a combination of highly accurate attitude rate gyros and linear accelerometers, which are arranged and processed in such a way that (ideally) the user can keep track of his position and attitude without external reference. An IRA is usually characterized by extremely good short-term stability (i.e., low noise), but it is also susceptible to long-term drift. There are two general forms of IRA, gimballed and strapdown.

A gimballed IRA consists of a set of rate gyros and accelerometers mounted on a platform, which is isolated from vehicle motion via a gimbal structure. The platform is usually kept aligned with an inertial or earth-fixed reference frame via torquing commands computed using the rate gyro and accelerometer outputs. The accelerometer outputs are integrated to provide linear velocity and position information in the chosen reference frame. Because the instrument platform rotates slowly and uniformly, the angular velocity environment is quite benign, and highly accurate relatively sensitive instruments can be used. The disadvantages of a gimballed IRA are: relatively high weight, volume, and cost; and vulnerability to single-point failures. A strapdown IRA is just what the name implies in that the rate gyros and accelerometers are firmly attached to the vehicle, and therefore are subject to all vehicle motion. The rate gyros must have a wide dynamic range to function properly in this environment, but also must be extremely accurate to allow the high-frequency processing of their outputs to track the attitude of the vehicle relative to a fixed reference. This attitude information is used with the accelerometer outputs to track the position and velocity of the IRA in the desired reference frame. In spite of the harsh dynamic environment of the strapdown system, laser gyros (with their absence of moving parts) have matured to navigation-level accuracies and stabilities in strapdown configurations.

Radio-navigation aids provide the user with position information with respect to known earth-based landmarks or earth satellites. In general, radio-navigation aids tend to be noisier in the short-term than an IRA, and susceptible to performance degradation due to vehicle maneuvers, but have essentially no long-term drift. In the recent past, many studies have successfully combined IRA and radio-navigation data in a near-optimal fashion using matched (Kalman) filters. Such systems utilize the long-term stability of the radio aids with the short-term stability of the IRA to provide extremely accurate position and velocity information.

In the time frame envisioned for the generic avionics system, two new radio-navigation aids will be available with accuracies superior to present systems. They are the NAVSTAR Global Positioning System (GPS) and the Joint Tactical Information Distribution System (JTIDS). A brief summary of these two systems follows; more detailed discussion is given in Section 6.

When finally complete, GPS will employ three rings of eight 12-hour earth satellites. Using redundant measurements to (at least) four satellites to accommodate user clock bias and drift, the GPS X-receiver set is expected to provide the user with earth-relative position fixes accurate to 22 feet (1 $\sigma$ ), and velocity fixes accurate to 0.1 foot/second (1 $\sigma$ ). These fixes can be provided at a 10-hertz rate.

JTIDS will be a secure multifunction information-exchange system with a two-dimensional navigation capability. The accuracies of the JTIDS navigation information (both absolute and relative to other members of the tactical community) are classified, but it can be said that this source is sufficiently accurate to be used for tactical navigation. Because of the many other types of data being transmitted, the update rate for JTIDS navigation information will undoubtedly be lower than that of GPS. However, due to the relatively short distances between members of the JTIDS community, the jam-resistance of JTIDS is expected to be significantly greater than that of GPS.

It is quite possible in a jamming environment that GPS will be able to provide position fixes, but not velocity fixes. This is because the two information types are obtained from different inner loops with different bandwidths, with the position fix being associated with the smaller bandwidth loop. Because satellite ephemeris data are also lost when velocity-fix information is lost, operation in this mode for



extended time periods would lead to degraded performance. However, utilizing the vast communication capability of JTIDS, a user outside the jamming environment (thus, receiving satellite ephemeris data) could send this information to the jammed GPS user via JTIDS, and large errors for that user could be avoided. This illustrates one of several ways in which GPS and JTIDS are complementary. (4-4)

To summarize, tradeoffs exist between gimballed and strapdown inertial reference assemblies, between GPS and JTIDS for radio navigation, and between an IRA and radio-navigation aids. The baseline avionics system can be chosen from among these elements only after a thorough analysis of these tradeoffs and their impact on total-system cost and performance.

The gimballed IRA offers computational advantages via direct-attitude angle readout and simple navigation equation mechanization in the inertial frame, but gimballed units tend to be heavier and more expensive than strapdown units. In addition, the loss of a single gyro in a gimballed IRA renders at least two accelerometers unusable because of the uncertainty in their orientations. Because of this inherent vulnerability of the gimballed IRA to single-point failures, it appears that a strapdown IRA is more appropriate to the generic avionics system, and (consequently) the gimballed IRA will not be discussed further.

GPS offers more frequent navigation updates than the JTIDS navigation function, but is less jam-resistant than JTIDS. An IRA offers self-contained navigation capability, but long-term drift stability can only be obtained with more expensive equipment, and some threshold drift is always present. On the other hand, radio-navigation aids offer bounded-error position and/or velocity measurements at moderate cost for the receivers, but are susceptible to jamming and do not have the dead-reckoning capability required to provide navigation information between navigation fixes.

#### 4.3.2 Flight-Control Instruments

Two general types of information have been utilized in the past for aircraft flight control: inertial information, relating aircraft motion to some inertial (or earth-fixed) reference system; and air data, indicating aircraft motion with respect to the air mass.

There are three classes of instruments that provide flight-control grade inertial information. These are strapdown attitude rate gyroscopes,

strapdown linear accelerometers, and attitude gyroscopes (i.e., vertical and directional gyroscopes). Although strapdown rate gyros and linear accelerometers have been discussed in the context of the IRA, the decreased accuracy requirements for flight control allow significantly less expensive instruments to be used for this function. While the gyros of the IRA provide accuracies of the order of 0.01 degree/hour, the angular rate accuracy required for flight control is of the order of 30 degrees/hour. However, if no attitude information (i.e., IRA or attitude gyros) is available, the rate-gyro information must be integrated to provide attitude information both for flight control and pilot display, and the required accuracy is raised to approximately 1 degree/hour (still two orders of magnitude less accurate than the IRA data). The 30-degree/hour accuracy figures can be met with spring-restrained gyroscopes, while the 1-degree/hour accuracy requires a gyroscope of the least expensive rate-integrating type, or better. It is important to note that, since the IRA rate gyros and accelerometers provide inertial information of much higher quality than required for flight control, the IRA can be used for flight control. If no inertial flight-control instruments are available except for the IRA, the flight-control function reliability requirement demands that the IRA reliability and survivability be consistent with flight-critical operation.

As mentioned previously, there are two instrument types (excluding IRA) that provide pilot display-grade attitude data: rate integrating gyros, and attitude gyros. The use of attitude gyros relaxes the accuracy requirement for the flight-control rate gyros, and the attitude gyros provide analytic redundancy that can be utilized to isolate rate-gyro failures, and vice-versa. A fundamental limitation of the analytic redundancy is that the size of drift that can be isolated in an attitude gyro is of the order of twice the nominal bias in an unfailed rate gyro. (4-5)

The fundamental importance of air data arises from the fact that the aerodynamic forces on the aircraft are directly related to the velocity of the aircraft with respect to the air mass. Aircraft inertial velocity and air-relative velocity differ by the inertial velocity of the air mass. The ultimate requirement for knowledge of airspeed in conventional horizontal flight is to avoid loss of lift, or stall, and large sideslip; and high-speed air data instrumentation is necessary for safe flight. At the extremely low groundspeeds encountered by V/STOL aircraft, the control forces are no longer generated by aerodynamic lift, but instead arise from mass expulsion; thus, aerodynamic



forces become disturbances that must be overcome by the control system. As an example, the AV-8 Harrier V/STOL aircraft is limited to less than a 10-knot crosswind in the vertical mode, and the Harrier must weather-vane into the wind in order to land successfully. (4-6) Because all-weather operation is an important requirement, successful takeoffs and landings must be made in adverse wind conditions. Thus, there is a tradeoff between the requirement for low-speed air data to allow the control system to orient the aircraft into the wind, and the requirement for low-speed controllability (in terms of excess thrust above weight, and roll-moment generating capability).

There are at least two proven approaches to obtaining high-speed (>60 knot) air data: the use of a multipurpose probe, or a combination of a conventional pitot/static probe with angle of attack and sideslip vanes. Multipurpose probes offer the advantages of compactness and minimum plumbing, while their disadvantages include rather complicated data processing to derive sideslip angle and angle of attack, and the loss of many measurements with the loss of a single probe. In contrast, the combination of a pitot/static probe with alpha and beta vanes has the advantages of less measurements lost with the loss of a single probe, and direct alpha and beta readout, which reduces computational complexity. The major disadvantage is the requirement for more aircraft surface area for mounting. Both the multiprobe and the pitot/static probe require measurement of total temperature to allow compensation of their outputs.

Reference 4-7 discusses several possible approaches to the low-speed air-data problem, and singles out two omnidirectional low-range airspeed systems as most promising. One system (called LORAS by its developers, Pacer Systems, Inc.) utilizes a 1-foot diameter arm, with pressure ports at either end, rotating in the airstream at 12 revolutions/minute. Using measurements of the pressure at the two ends of the arm, the airspeed magnitude and direction in the plane of rotation can be calculated, and operational tests below 130 knots have demonstrated measurement accuracies of approximately 3 knots. The LORAS system can operate at speeds up to 250 knots, and can withstand supersonic speeds when turned off. The other system (designed by Rosemount) utilizes a probe very similar to the multipurpose probe discussed earlier, but mounted orthogonal to the aircraft longitudinal axis. By mathematical manipulation of the pressures measured in four chambers in the probe, the airspeed magnitude and direction in the plane normal to the

probe can be calculated. The Rosemount sensor is stated in Reference 4-7 to operate at speeds up to 50 knots, although Reference 4-3 indicates that operation at speeds up to 120 knots is possible. Tests below 50 knots have demonstrated accuracies within 5 knots. Note that two sensors of either the Rosemount or Pacer design are required to allow calculation of the full three-dimensional airspeed vector.

The use of pressure transducers distributed over the surface of the aircraft is a radically different approach to low-speed air data, and offers the possibility of directly measuring the quantities of interest, namely lift and sideforce. Although this is an unproven technique, and the redundancy-management approach for such a system is unclear in the face of localized pressure effects due to flow pattern changes, it is felt that the technique should be investigated in the future.

The automatic-landing task naturally divides into long-range and short-range portions. The short-range task involves the actual landing of the aircraft at the landing site, and its most severe test is ship-board landing of a V/STOL in Sea State 5. The long-range task involves an intermediate phase during which the aircraft is positioned to enable the acquisition of the short-range landing system (SRLS) signals.

For the long-range task, a microwave landing system (MLS) receiver could provide three-dimensional relative-position information. Information of similar accuracy could be provided by a combination of a GPS-aided IRA and JTIDS, with the aircraft receiving the ship's inertial position and velocity through JTIDS, and then computing relative position and velocity by subtracting its GPS-aided IRA data.

For all weather operation of a V/STOL aircraft from small ships, a high level of automation will be necessary to assure consistently safe landings. Even with landing aids, it appears doubtful that in high seas the task can be performed totally manually, and an automatic short-range landing system may be necessary for safe operation.

Although these factors are significant in and of themselves, there is an additional important reason to pursue the technology required for automatic V/STOL landing. In vertical flight, a V/STOL aircraft expends fuel at an extremely high rate. Typically, the fuel required for vertical landing must be carried through the entire mission; hence, it strongly impacts the overall aircraft design and especially its gross weight. If V/STOL landings can be accomplished in minimum



time, effectively eliminating the long hover periods that usually occur in manual landings, very significant savings in vehicle gross weight can accrue, with corresponding savings in life-cycle costs. Thus, an automatic-landing system, which can support high-deceleration, positive, safe landings in all-weather conditions, can have a potentially enormous impact on the viability of V/STOL aircraft.

Some important requirements should be satisfied by such an SRLS. Since the aircraft will be required to land on either sea-based or land-based platforms (especially in forward-land-based areas), the ground-based equipment should be minimal. Thus, the airborne system should be mostly self-contained, requiring little more than a portable transponder at the land-based landing site.

A variant of this transponder could be used to indicate ship position, relative to the aircraft, for sea-based landings. However, since the aircraft must be captured aboard the heaving rolling ship, a positive arresting mechanism will be necessary to capture the aircraft on touchdown. It is reasonable to expect that this mechanism will require a close match between ship and aircraft relative velocity, attitude, and attitude rate at touchdown. In effect, the aircraft must match ship motion to within reasonable tolerances at touchdown. Thus, during its approach, the aircraft will require ship velocity, acceleration, attitude and attitude-rate information. The necessary information is all available from a shipboard inertial navigator, and can be transmitted via radio link to the aircraft.

Another important element of such a system is security. Since it has the potential to divulge both ship and aircraft position to enemy forces, the automatic-landing system must employ appropriate means to thwart direction finding and other detection methods. Probably the most effective means is to employ appropriate frequencies and low power, to limit the effective broadcast range of the SRLS.

#### 4.4 Failure Detection and Isolation Techniques

Instrument failure detection and isolation is an integral portion of the generic fault-tolerant avionics system. It has a significant impact on redundancy levels, and hence the design of the avionics system. The requirement for reliable isolation and removal of failed flight-control sensors from flight-critical control calculations is readily apparent. Automatic instrument FDI also results in significant savings in maintenance troubleshooting costs by pinpointing the

failed line-replaceable unit. In addition, because the form of FDI used can influence the types and number of instruments necessary for a stipulated level of fault-tolerance, it follows that only through an integrated comprehensive FDI approach can the requirements of the avionics system be met in a manner that is cost-effective over the life of the aircraft. It is crucial that the FDI system provide levels of performance in terms of acceptably small false-alarm and missed-alarm probabilities; hence, a systematic approach, well founded in theory and experience, should be employed.

This section contains a discussion of three general methodologies for accomplishing FDI: (1) self-test, (2) direct redundancy, and (3) analytic redundancy. Self-test utilizes any of a variety of monitoring techniques to isolate the failure of an instrument, using only the behavior of the instrument in question. Direct redundancy utilizes voting among the outputs of like sensors to isolate the failed sensor, while analytic redundancy utilizes the outputs of unlike sensors to isolate the failed instrument.

#### 4.4.1 Self-Test

Self-test, as the name implies, relies upon information that can be obtained within the unit itself to isolate a failure. Several techniques for self-test have been used in the past, including signal wrap-around comparisons to monitor digital/analog converter performance, the use of small-amplitude test signals to monitor input/output performance, and software reasonability checks on successive instrument outputs. In the generic avionics system, the availability of embedded microprocessors in instrument modules will greatly facilitate self-test.

As discussed in the following, self-test offers high potential benefits for laser-gyro fault isolation, and similar benefits may be obtained for other sensor types as well. It is important to note that the hardware self-test, or built-in test equipment (BITE), envisioned for the generic avionics system will differ significantly from past BITE efforts, which have universally been treated as secondary minimum-cost minimum-precision designs. As a consequence of this philosophy, most previous BITE systems have had high false-alarm probabilities. Since a false BITE alarm results in the loss of the use of that instrument, it affects the avionics system in the same manner as an actual instrument failure. Therefore, BITE false-alarm probabilities must be appreciably lower than the instrument failure probabilities in order



for BITE to be useful. Additionally, unless the BITE probability of identifying a failure (failure coverage) is extremely high, there is little justification for BITE at all, even with low false-alarm probability, since an alternate form of failure isolation must still be provided. Several possible areas of investigation for improving BITE effectiveness are as follows:

- (1) Extend BITE failure coverage using failure mode and effects analyses, such as those being conducted in the Integrated Inertial Sensor Assembly (IISA) Program.
- (2) Examine the effectiveness of redundant BITE hardware.
- (3) Examine the use of intelligent or adaptive BITE thresholds using microprocessor-driven logic.
- (4) For the laser gyro application, monitor analog output signals associated with path-length control and discharge current for use in scale-factor mode shift sensing and trend reporting.

If the failure coverage of self-test, through both hardware and software mechanizations, can be raised to a suitably high level with accompanying low false-alarm probabilities, then a minimally redundant complement of sensors could be realized, with only the redundancy necessary to meet the failure tolerance requirements and no instruments added merely to facilitate fault isolation. To illustrate this point, only two instruments must be provided to allow the measurement of a scalar following a single instrument failure (i.e., single fault tolerant) if self-test can be used for failure isolation, but three instruments must be provided if direct-redundancy failure isolation (voting) is required.

#### 4.4.2 Direct-Redundancy Tests

Direct-redundancy tests utilize the outputs of like sensors to detect and isolate an instrument failure. As mentioned earlier, at least  $n + 1$  instruments measuring different components of an  $n$ -dimensional vector must be present to detect one instrument failure, and at least  $n + 2$  instruments must be present to isolate the failed sensor. Most direct-redundancy tests use as decision variables linear combinations of the like instrument outputs. These linear combinations (or parity equations) are defined in such a way that, due to the geometry of the instrument measurement axes, they only involve the instrument errors and

are independent of the value of the variable being measured. A simple illustration of direct-redundancy failure isolation using parity equations is the case of three measurements of a scalar. Three parity equations can be defined as follows:

Equation (1): The output of Instrument 1 minus the output of Instrument 2.

Equation (2): The output of Instrument 1 minus the output of Instrument 3.

Equation (3): The output of Instrument 2 minus the output of Instrument 3.

If the magnitude of any one of the equations is large, a failure is detected. Instrument 1 is identified as failed if the magnitudes of equations (1) and (2) are both large; Instrument 2 is identified as failed if the magnitudes of equations (1) and (3) are both large; and Instrument 3 is identified as failed if the magnitudes of equations (2) and (3) are both large.

Several FDI techniques utilizing direct redundancy have been formulated for arrays of sensors measuring components of a three-dimensional vector, some working directly with parity equations and others not. In Reference 4-8, a method is developed for strapdown rate-gyro and accelerometer FDI in which a strict definition of a failed instrument is used; i.e., an instrument having an error magnitude greater than a stipulated value. In this technique, the parity equation coefficients and a threshold are defined consistent with the failed instrument definition, such that a failure is detected if any parity equation is of greater magnitude than the threshold. An instrument is identified as failed if it is not a member of any parity equation of magnitude less than the threshold. Assuming only one instrument is failed (i.e., has an error magnitude above the stipulated value), this technique is guaranteed to give no erroneous detections or identifications. Reference 4-9 describes the Redundant Strapdown Inertial Reference Unit (SIRU) System, in which instrument FDI utilizes the least-squares estimate of the measured quantity, either linear acceleration or angular rate. The least-squares instrument residuals are calculated as the differences between the actual instrument readings and their theoretical readings if the quantities being measured were equal to the least-squares estimates. An instrument failure is detected when the total squared error



(TSE—the sum of the squares of all residuals) exceeds a detection threshold. An instrument is identified as failed when the square of its residual exceeds the product of an isolation threshold times the TSE. In the SIRU study, the detection and isolation thresholds were selected empirically; however, recent papers\*, utilizing the concept of "parity space", describe an analytic technique whereby the detection and isolation thresholds may be determined by calculating missed-alarm, false-alarm, and misidentification probabilities as functions of the thresholds. Once these functions are evaluated, the thresholds chosen are those that give the desired performance.

The concept of parity space FDI can be illustrated using the aforementioned example of the measurement of a scalar with three instruments. Three possible parity equations for such a system were given. However, the parity space technique recognizes that only two of those equations are linearly independent (e.g., equation (3) is equal to equation (2) minus equation (1)), and the dimension of the parity space in which FDI calculations are performed is 2. In general, for  $m$  instruments measuring the components of an  $n$ -dimensional vector, parity space is of dimension  $(m - n)$ . A matrix of parity equation coefficients may be defined such that the noises in the parity vectors due to normal output errors in unfailed instruments are independent for the case of independent instrument noise, allowing analytic determination of the performance properties of the FDI technique. Because the failure of any given instrument results in a parity vector with a prescribed orientation in parity space, the use of this technique in those cases where the excess number of instruments over the dimension of the measured quantity is three or less results in simple geometric interpretations of the FDI process. It should be mentioned, however, that at this time the power of the parity space method seems to lie in performance analysis and not in algorithm design per se.

It is important to emphasize that if like instruments are spatially separated, and therefore relative motion among the instruments is possible, then (in the strictest sense), direct redundancy no longer exists, but is replaced by analytic redundancy. The degree to which this impacts an FDI scheme is in direct proportion to the amount of relative motion possible, scaled by the nominal instrument error characteristics. For the case of flight-control-level bias failures in

---

\* See References 4-10, 4-11, and 4-12.

spatially separated inertial instruments on an aircraft, however, body-bending can probably be accommodated in a direct-redundancy scheme by one of two related methods:

- (1) Averaging the parity equations over an interval containing many bending cycles.
- (2) Filtering the bending frequencies out of the sensor outputs before forming the parity equations.

On the other hand, extensive analyses of the Space Shuttle have demonstrated that comparison of individual outputs from spatially separated IRAs is insufficiently accurate to allow reliable isolation of navigation-level failures.<sup>(4-13)</sup>

#### 4.4.3 Analytic-Redundancy Tests

The term analytic redundancy refers to the redundancy derived from functional and kinematic relationships among variables measured by unlike instruments. Because, like self-test, analytic redundancy offers the potential for a minimally redundant sensor complement, the full potential of analytic redundancy should be investigated before the final configuration of the generic avionics system is defined. The following paragraphs provide a summary of the analytic-redundancy methods that may be applicable to the generic avionics system. Although only sensor failure isolation is discussed, analytic redundancy offers great promise for effector failure isolation as well.

##### 4.4.3.1 Sequential Tests

It is proposed that FDI using analytic redundancy be implemented in sequential versus single-sample tests since, on the average, sequential tests significantly outperform fixed-interval tests. A particularly effective sequential test, that has been applied with excellent results in the F-8 digital-fly-by-wire (DFBW) analytic-redundancy-management program,<sup>(4-5, 4-14)</sup> is the Modified Sequential Probability Ratio Test (MSPRT). The MSPRT is based upon Wald's SPRT,<sup>(4-15)</sup> a binary hypothesis test that decides whether or not the observed noisy process contains a stipulated mean value. The thresholds of the test are determined by the two stipulated acceptable probabilities of choosing the wrong hypothesis. One technique for failure isolation using the SPRT is to define a residual process for each suspect instrument as the difference between a convenient function of its output and



an equivalent expression calculated using other instruments in the complement. (Details of these residual processes for the various instrument types are discussed in the following.) A mean for the SPRT is defined as the residual signature of a bias of a predefined size, implying that any instrument having an error of this magnitude or larger is considered failed. The sign of the SPRT mean is determined from the sign of a parity equation, and if no parity equation is available, two SPRTs must be employed. The failed instrument is identified as the one whose SPRT first indicates that the bias failure mean is present in its residual.

It is important to note that the residual processes employed here, defined as the difference between analytically equal expressions, are in contrast to those used in some previous approaches in which all redundant information is fed into a single filter and the residuals from this filter are used for FDI. In essence the proposed approach represents a zero-gain or open-loop filter, and increases the failure observability over the aforementioned nonzero-gain super-filter approach.

Although analytic redundancy has the potential to reduce component redundancy, by its very nature, analytic-redundancy residuals are noisier than direct-redundancy residuals, and (in particular) analytic-redundancy residuals can contain low-frequency error terms due to such effects as modeling errors in the analytic relationships and nominal instrument error characteristics. In order to accommodate such effects, while still retaining the simple structure of the SPRT and its inherent optimal properties,<sup>(4-16)</sup> the MSPRT can be utilized. The MSPRT is a minimax approach in which the effects of postulated worst-case error terms (not due to instrument failure) are accommodated as SPRT threshold offsets. Compared with SPRT performance, as long as the actual erroneous terms are smaller than the postulated values, the misclassification probabilities using the MSPRT will be no greater than those used to define the original SPRT thresholds, but the mean time to isolate a given failure using the MSPRT will be longer than that using the original SPRT. The use of the MSPRT is preferable to a fixed ad hoc delay imposed upon the SPRT, since the MSPRT error terms are often maneuver-dependent, and the use of a fixed delay to accommodate the largest possible error would be prohibitively conservative. In order to decrease the effect of the delay caused by the threshold offset, the concept of provisional failure isolation has proven useful in the

case of redundant sensors. In this approach, the output of an instrument whose MSPRT is tending toward the failure decision is removed from all critical calculations pending final MSPRT threshold crossing.

It is important to note that the MSPRT approach is applicable to direct-redundancy residuals as well, and its use of threshold offset seems particularly well suited to false-alarm prevention during times when unfailed-instrument parity equation magnitudes might be high; e.g., due to nominal gyro scale-factor errors in high-rate high-acceleration maneuvers.

#### 4.4.3.2 Analytic Redundancy for Flight-Control Instruments

The instruments enumerated earlier as applicable to the flight-control function are linear accelerometers, attitude rate gyros, attitude gyros, air data sensors, and automatic-landing system receivers. In this section, the various forms of analytic relationships available for these sensor types are outlined, together with the associated MSPRT means and worst-case error sources.

4.4.3.2.1 Inertial Instruments and Air Data.—The outputs of the attitude gyros and attitude rate gyros are related through nonlinear differential equations. Analytic-redundancy residuals are defined using numerical integration of these differential equations, with the result that the residual signature for a bias in a rate gyro is a ramp, and the residual signature of a bias failure in an attitude gyro is a step. The dominant error sources in these residuals are as follows:

- (1) Initial attitude gyro noise.
- (2) Rate-gyro axis misalignment.
- (3) Nominal rate-gyro bias.
- (4) Nominal rate-gyro scale-factor error.

It should be noted that although these analytic-redundancy relationships between attitude gyros and attitude rate gyros, referred to as rotational kinematics (RK), are exploited to great advantage using the instruments aboard the F-8 DFBW aircraft, the rate gyros aboard the aircraft are of such a grade that identification of bias levels of the order of 10,000 degrees/hour is appropriate. It remains to be demonstrated whether the inherent increase in mean time necessary to identify gyro biases of the accuracy required for V/STOL flight control (30 degrees/hour) would render this form of analytic redundancy unusable.



The derivative of the air-relative velocity of the aircraft is a function of variables measured by the linear accelerometers, air data sensors, rate gyros, and vertical gyros. Using residuals calculated by numerical integration of this equation, this form of analytic redundancy, referred to as translational kinematics (TK), can be used to isolate failures in these instruments. The following comments can be made on TK, based upon the F-8 DFBW experience:

- (1) The major low-frequency error source is the wind acceleration, which cannot be estimated in the presence of an instrument failure. The method used in the F-8 to accommodate this unknown modeling error is the use of a threshold offset arising from the effect of a constant-magnitude wind acceleration for 12 seconds, changing sign at 6 seconds. The magnitude of the wind acceleration used assumes one of two values depending upon the estimated turbulence level. Using this approach, a failure equivalent to a 150 mg accelerometer bias requires a mean detection time of the order of 13 seconds.\*
- (2) The signature of a rate-gyro bias failure is modulated by the airspeed, and no signature is given if the rate-gyro input axis is parallel to the aircraft velocity vector. Assuming the failed rate-gyro axis is normal to the velocity vector, an airspeed of approximately 600 feet/second is required for a 1600 degree/hour rate-gyro bias to be equivalent to a 150 mg error in computer airspeed rate.
- (3) The use of TK for vertical-gyro failure isolation requires bias jumps of approximately 9 degrees to produce 150 mg errors in computed airspeed rate.
- (4) The observability of a bias in the measurement of  $\alpha$ ,  $\beta$ , or angular rate is proportional to airspeed, and is essentially zero below 60 knots.

Relationships between aircraft acceleration and models of aerodynamic force utilizing air data measurements are referred to as translational dynamics (TD) analytic redundancy. On the F-8, the use of TD

---

\* It is important to note that for all MSPRTs, the isolation times for failures larger than the postulated value are proportionately shorter than those quoted for failures of the postulated size.

yields reliable isolation of lateral accelerometer biases of 150 mg, and alpha vane biases of 2 degrees, at Mach numbers above 0.4. That implementation utilizes polynomial representations for the aerodynamic forces as high as fourth order, and requires approximately 200 polynomial coefficients to represent the aircraft in the clean configuration. The following comments are relevant to the applicability of TD to the FDI problem:

- (1) The primary error sources in the TD equations are: lack of knowledge of the true aerodynamic-force characteristics, nominal biases in the air data measurements, and lack of knowledge of propulsive thrust. The expected magnitudes of these errors limit the instrument failures that can be isolated.
- (2) In the TD residuals, the observability of a Mach bias is proportional to airspeed, and the observability of an  $\alpha$  or  $\beta$  bias is proportional to the square of airspeed, limiting isolation at low to moderate airspeeds.
- (3) Because of the probable skewed orientation of the flight-control accelerometers, isolation of failures in these instruments via TD will require the evaluation of the complete three-dimensional aerodynamic and propulsive force vectors at each sample time.

Another form of analytic redundancy between flight-control sensors, called altitude kinematics (AK), arises from the fact that the vertical inertial acceleration is equal to the second derivative of altitude—a function of measured acceleration and attitude. Altitude kinematics provides redundancy between measurements of altitude, either using static pressure and temperature data or a radar altimeter, and the linear accelerometers and vertical gyros. The AK test is effective in isolating failures only in accelerometers with input axes within 20 degrees of the vertical. It is effective in isolating failures in altitude measurements, but requires the storage of a residual window. If only the measurement of altitude, and not altitude rate, is available, a filter for each altitude device is required for altitude rate estimation. Error in the knowledge of altitude rate at the time of failure detection is the dominant low-frequency error in AK residual calculations.



An additional form of flight-control inertial-sensor failure isolation will be available, since both the flight-control linear accelerometers and angular rate gyros will have counterparts in the IRA of much higher quality. Although it is probable that the IRA will be spatially separated from the flight-control sensors, introducing relative motion (and therefore analytic, and not direct redundancy), it is felt that flight-control level failure isolation can be accomplished using sequential direct-redundancy tests with high-frequency errors removed by increased test variance or notch-filtering the instrument outputs. This form of redundancy should be sufficient for flight-control accelerometer and rate-gyro FDI as long as the IRA is operating.

Geographic kinematics (GK) refers to the redundancy between the position and velocity estimates obtained using onboard inertial instruments and the position and velocity estimates obtained from radio-navigation aids (in particular, GPS and JTIDS). The major error sources for GK tests are as follows:

- (1) Nominal unfailed sensor bias.
- (2) Nominal unfailed sensor scale-factor error.
- (3) Nominal radio-navigation measurement bias.

Table 4-1 gives a summary of the predicted behavior of GK using GPS position or velocity data in isolating failures of flight-control rate gyros and accelerometers. These results assume a GPS update rate of 10 hertz. The attitude error figures assume that the IRA has been lost due to battle damage, and only a single attitude equation, utilizing all rate-gyro outputs, is employed. It is also assumed that the aircraft is not maneuvering, and that the input axis of a failed rate gyro is normal to the specific-force vector. The errors listed in Table 4-1 as being accommodated are used to calculate the MSPRT threshold offsets, and the position and velocity errors reflect predicted GPS performance.

Two important points should be made about GK for flight-control-level inertial-instrument FDI. First, the GPS velocity signals would be the first to be lost in a jamming environment, and therefore it would be prudent to design the GK tests using the GPS position data. However, the velocity bias accommodated in Table 4-1 (0.005 foot/second) reflects only nominal GPS velocity bias. If the expected velocity bias from a combined flight-control/GPS/JTIDS navigator (implemented following loss of the IRA) were significantly larger than this figure, the

mean isolation times in the last three rows of Table 4-1 would be optimistic, and new results accommodating the larger velocity bias would be required. The second point is that, while Table 4-1 indicates that isolation of 10 mg accelerometer biases is feasible using GK, it also indicates that prohibitively long times are required to isolate 1 degree/hour rate-gyro failures, if accelerometer errors as high as 10 mg must be accommodated.

Table 4-1. Flight-control sensor failure isolation via GK using GPS.

Failure Magnitude	Errors Explicitly Accommodated	Type of GPS Data	Mean Isolation Time (s)	Resulting Attitude Error (deg)
1 deg/h	10 mg, 0.005 ft/s	vel	10000	2.8
1 deg/h	50 $\mu$ g, 0.005 ft/s	vel	54	0.015
10 mg	1 deg/h, 0.005 ft/s	vel	0.6	—
1 deg/h	10 mg, 0.005 ft/s, 15 ft	pos	14000	3.9
1 deg/h	50 $\mu$ g, 0.005 ft/s, 15 ft	pos	155	0.04
10 mg	1 deg/h, 0.005 ft/s, 15 ft	pos	58	—

4.4.3.2.2 Autoland Receivers.—The MLS receiver provides range and heading information to the aircraft relative to the landing site. A GK MSPRT can be utilized to identify MLS failures by observing the process defined as the change in MLS range and heading data minus the change in position information from the combined IRA/GPS navigator.

The proposed SRLS utilizes a transmitter/receiver on the aircraft that sends out and receives pulses, and a transponder at the landing site that retransmits the received pulses from the aircraft, providing range, azimuth, and elevation to the landing site during the terminal landing phase. In addition, for the case of landing on a ship, the shipboard transponder inserts ship-motion information between the retransmitted pulses, and this is the only high-frequency ship-motion information available to the aircraft. The translational ship-motion information from the SRLS, together with the range, azimuth, and elevation information can be used in a GK test with IRA/GPS information for isolation of an SRLS failure. Unfortunately, there is no source of ship attitude information except the SRLS. Although it is



possible that JTIDS could be used as an alternate communication path to provide low-frequency ship attitude information, comparison of these data with SRLS output would be difficult. It may be possible to perform coarse dynamic consistency checks on the SRLS ship attitude motion data through the use of a ship model in the aircraft software, but it is not clear at this time how useful such a check would be.

#### 4.4.3.3 Analytic-Redundancy for Navigation Instruments

The GK analytic redundancy described earlier for flight-control inertial-sensor FDI can also be utilized for isolation of failures in the IRA gyros, or accelerometers, or GPS or JTIDS receivers. GPS and JTIDS information can be used to isolate IRA failures, and IRA measurements can be used to isolate failures in GPS or the navigation function of JTIDS. Table 4-2 summarizes the predicted performance of GK tests in isolating an IRA rate-gyro failure, an IRA accelerometer failure, and a GPS velocity measurement failure.

Table 4-2. Navigation-instrument failure isolation using GK.

Failure Magnitude	Errors Explicitly Accommodated	Type of GPS Data	Mean Isolation Time (s)	Resulting Attitude Error (deg)
0.01 deg/h	50 $\mu$ g, 0.005 ft/s, 15 ft	pos	7011	0.02
50 $\mu$ g	0.01 deg/h, 0.005 ft/s, 15 ft	pos	251	—
0.1 ft/s	50 $\mu$ g	vel	1.9	—

#### 4.5 Major Tradeoffs and Conclusions

Because of the uncertainty associated with the fault tolerance and survivability requirements for aircraft that may utilize the generic avionics system, and because the reliability of much of the aforementioned avionics equipment in the 1990s is conjectural, it is difficult at this time to completely define a baseline generic avionics system. However, in most function areas, clear tradeoffs do exist between equipment types, equipment replication, and analytic-redundancy failure-isolation techniques. A thorough understanding of these tradeoffs is essential to effective baseline selection, and this section enumerates the various tradeoffs and the conclusions that can be drawn from them at this time.

#### 4.5.1 Inertial Components (Navigation and Flight Control)

Because of the flight-control capability of the IRA, the choice of inertial instruments to simultaneously fulfill the requirements for flight control and navigation must be made together. Table 4-3 indicates the fault-tolerance of several possible configurations of flight-control-grade (FC) and navigation-grade (NAV) accelerometers or rate gyros. Because the weight penalty associated with armor plating for survivability seems unwarranted, while the use of other equipment for shielding or placement near the pilot seems tenuous and cavalier at best, survivability of the flight-control function is achieved by spatial separation. In Configuration 1, a bulkhead divides two triads of navigation-grade instruments, with no flight-control-grade instruments present. In Configurations 2 through 6, spatial separation of the order of a meter is present between the clustered instruments of the IRA and the flight-control instruments. Although from the sensor-alignment, accessibility, and redundancy-management points of view it would be preferable to cluster the flight-control instruments in Configurations 2 through 6; additional flight-control function survivability can be attained by spatially separating these instruments. By judicious

Table 4-3. Fault-tolerance of various configurations.

Config- uration	Number of skewed accel- erometers or rate gyros		Navigation function fault toler- ance without damage		Flight-control function fault tolerance without damage		Flight-control function fault tolerance with damage to all navigation sensors in IRA cluster	
	NAV	FC	DR	AR	DR	AR	DR	AR
1	6*	0	2	3	2	3	0	0
2	4	4	0	1	4	5	0	1
3	5	5	1	2	6	7	1	2
4	5	4	1	2	5	6	0	1
5	5	3	1	2	4	5	0	0
6	6	6	2	3	7	8	2	3

\* Three instruments on either side of a bulkhead



mounting of the instruments at various locations within the fuselage (perhaps at various locations on a single bulkhead, as suggested in Reference 4-3), it should be possible to keep the relative motion between the sensors acceptably small for redundancy-management purposes.

In Table 4-3, fault isolation is achieved using either direct redundancy (DR) alone, or by a combination of direct redundancy plus analytic redundancy (AR). Self-test is not assumed because of its instrument-specific nature. The fault-isolation capability for the flight-control function via direct redundancy assumes the use of the navigation sensors (when available) for this purpose. The major trade-off (apparent in Table 4-3) for Configurations 2 through 6 is the large degree of flight-control-function fault tolerance provided in the absence of damage in order to assure survival of the flight-control function following limited battle damage.\* Although Configuration 1 suggests an approach that removes excessive flight-control fault tolerance, it is not clear at this time, whether (in fact) the approach will leave these instruments intact following limited battle damage (e.g., fire on one side of the bulkhead), or whether the instruments on the two sides of the bulkhead could have sufficiently accurate relative alignment. Thus, two questions, which must be answered before an inertial-sensor baseline configuration can be chosen, are as follows:

- (1) Is flight-control-function damage tolerance through spatial separation a requirement?
- (2) Can that requirement be met by an arrangement involving separation by bulkhead thickness?

The figures in Table 4-3 that indicate additional fault tolerance via analytic redundancy presume that unjammed GPS signals are available to allow the use of GK tests. It is an interesting paradox that the IRA requires the unjammed GPS signals to isolate low-level failures of its instruments, but that in such an unjammed situation this high accuracy is not required since GPS aiding of the IRA outputs will prohibit any excessive navigation errors. An important question that must be answered before designing the total navigation system is:

---

\* A limited-damage event is assumed to destroy all instruments in the IRA cluster, with damage localized to that area.

"What kind of a failure is the jamming of the GPS receivers? In particular, is it considered a failure at all in an 'n-failure operational' design?"

It is important to reiterate that for a configuration using pilot-display-grade integrating rate gyroscopes with no attitude gyros, isolation of a 1 degree/hour rate-gyro bias using GK analytic redundancy requires a prohibitively long isolation time when an accelerometer bias of 10 mg must be accommodated. If the level of attitude error before failure isolation or the long time period of failure-isolation processing were unacceptable, one of the following steps would have to be taken:

- (1) Use of flight-control accelerometers with significantly better accuracy than 10 mg. The cost difference between the two accelerometer types would determine the efficacy of this approach.
- (2) Augment the flight-control sensor complement by an additional rate gyro. This approach seems most cost-effective.
- (3) Addition of attitude gyros. This alternative does not seem acceptable, and is discussed in the following.

The use of attitude gyros in the generic avionics system does not seem necessary or cost-effective at this time, regardless of the chosen configuration of rate gyros and accelerometers. The addition of a different sensor type or types (at least one vertical gyro (VG) and one directional gyro (DG), or a single three-axis instrument would be required) would result in significant logistics costs over the system lifetime; and it is felt that these costs, plus the attitude-gyro acquisition cost, would be much greater than the acquisition cost of an additional rate gyro to provide one more level of direct redundancy.

#### 4.5.2 Air Data

Because of the flight-critical nature of loss of lift or excessive sideslip angle in many regions of an aircraft's flight envelope, some form of air data instrumentation (both high and low speed) will undoubtedly be required for the generic avionics system. For high-speed air data, the choices seem to be limited to either multipurpose probes or a combination of pitot/static probes and vanes. The redundancy-management techniques for either approach would be similar, and the



ramifications of the choice of one approach versus the other are relatively small. On the other hand, there are two very dissimilar approaches to low-speed air data instrumentation: an omnidirectional low-range airspeed system, and distributed pressure transducers.

The most promising candidate of the former approach is LORAS, since it has demonstrated highly accurate performance and good maintainability. However, the accuracy of direct redundancy for widely dispersed low-range airspeed instruments is questionable in the face of local airflow effects. Multiple LORAS units would require some separation for physical clearance of the rotating booms, and to minimize the interaction of air motion induced by one rotating sensor on the flow measured by another sensor. Because the Rosemount omnidirectional sensor is similar to a multipurpose probe and does not rotate, the mounting of several of these sensors in close proximity should not create any major interference effects. Therefore, it is proposed that extensive testing be performed on the Rosemount sensor to determine whether its design can be matured to the level of LORAS. Both LORAS and the Rosemount sensor measure airspeed and direction in a plane, and two instruments of either type are required to measure the full airspeed vector. Since there is no alternate source of low-speed air data in addition to these systems, FDI for these instruments must be accomplished using direct-redundancy tests, and the technique would be similar to that employed for FDI for two-degree-of-freedom rate gyros. (4-12)

The concept of distributed pressure transducers to provide direct measurement of lift and sideforce is appealing because these are the quantities of interest for flight control. However, the technique is unproven, and the redundancy-management approach for these sensors is unclear at the present time, due to the potentially large local variations in the air flow at distributed locations on the wing, rudder, and fuselage. Nevertheless, the area of reliable low-speed air data from distributed pressure transducers appears to be a promising one for research.

#### 4.5.3 Radio-Navigation Aids

Because of the extremely accurate geographic position available from GPS and the highly jam-resistant communications channels provided by JTIDS, it is likely that at least one GPS receiver and one JTIDS receiver will be onboard the aircraft. Beyond this minimum complement,

it will likely be necessary to include a second JTIDS receiver to provide secure redundant communication following the failure of one receiver, relying upon self-test for the isolation of the failed unit. If, in the future, JTIDS self-test is found deficient in fault-isolation coverage, and guaranteed automatic JTIDS operation following a single failure is still required, the utility of GK analytic redundancy with GPS/IRA data to isolate JTIDS failures should be investigated.

Because of the jam susceptibility and relatively high projected cost of a GPS receiver, it appears at this time that not more than a single GPS receiver should be in the avionics complement. If, as envisioned, the IRA instruments are of sufficient accuracy to satisfy navigation-function requirements in an unfailed stand-alone mode, the cost tradeoff for IRA fault tolerance between an additional laser gyro and accelerometer for direct redundancy, versus an additional GPS receiver for analytic redundancy, will undoubtedly favor the added IRA instruments. On the other hand, if for some reason it is required that GPS capability be retained after the failure of one receiver, two GPS receivers must be onboard, with failure detection accomplished using direct redundancy, and failure isolation accomplished using GK. In this case, the additional IRA instruments would not be necessary.

#### 4.5.4 Autoland Receivers

Because of the likely requirement that the SRLS transceivers be fail-operational after initiating the landing sequence, and because there is no redundant information available regarding ship motion, it appears necessary that there be three SRLS transceivers aboard the aircraft. If future development of the SRLS transceiver indicates it has a sufficiently long mean time between failures (MTBF), the short exposure time of the landing sequence might allow removal of the fault-tolerance requirement, and only two SRLS transceivers would be needed. In that case, the landing sequence would not be initiated if the readings from the two transceivers disagreed.

Because there is a good source of redundant MLS information from IRA/GPS, three MLS receivers appear unnecessary to provide single failure tolerance. Thus, the inclusion of two MLS receivers in the aircraft can provide single-failure tolerance, with the failure detected using direct redundancy and isolated using the IRA/GPS navigation data.



#### 4.6 Summary and Recommendations

In the preceding sections, the requirements of the generic avionics system have been discussed, together with existing and potential equipment to satisfy the functional requirements, and available FDI techniques to satisfy possible fault-tolerance and survivability requirements. The complete stipulation of a baseline avionics system is difficult at the present time for the following reasons:

- (1) The fault-tolerance and survivability requirements are currently unspecified. To a great extent, the fault-tolerance requirements will be dependent upon the MTBFs of the equipment types in the complement; and these figures are difficult to estimate for the 1990s time frame. This is especially true for new design equipment.
- (2) A number of promising research areas exist that could impact the choice of sensor complement. These will be discussed later in this section.
- (3) The acquisition costs for the different sensors are not easily estimated, making tradeoffs between equipment types difficult to assess.

In spite of the aforementioned difficulties, the various trade-offs discussed in Section 4.5, together with some engineering judgments concerning fault-tolerance and survivability requirements, do suggest a tentative baseline avionics system, summarized in Table 4-4. Although the redundancy of a particular instrument type may change between the value in the table and the baseline value, it is not anticipated that major changes will occur. However, it is likely that some instrument types (such as Doppler radars or correlation velocity sensors, and radar altimeters) will be added to satisfy various mission-related requirements.

The IRA for the tentative baseline design consists of four laser gyros and four accelerometers. It is anticipated that these instruments would be clustered, with their input axes normal to the faces of an octahedron. The tentative parameters for these instrument types are given in Tables 4-5 and 4-6. The IRA is required to perform self-alignment, with level misalignments of the order of 20 seconds of arc (1σ), and heading misalignments of the order of 150 seconds of arc (1σ). This requirement dictates accelerometer alignment accuracies of approximately 10 seconds of arc (1σ) and bias stability of 50 μg (1σ).

Table 4-4. Tentative core avionics sensor system baseline.

Type	Quantity
Laser Gyros	4
Accelerometers (Navigation Grade) } IRA	4
Rate Gyros (Display Grade)	4
Accelerometers (Display Grade)	4
Multipurpose Air Data Probes	2
Temperature Sensors	3
Onmidirectional Low-Range Airspeed Systems (V/STOL only)	3
GPS Receivers	1
JTIDS Receivers	2
Microwave Landing System Receivers	2
Short-Range Landing System Transceivers (V/STOL only)	3

Table 4-5. Laser-gyro parameters.

Scale Factor	1.5 s/pulse	Assumed stable over life of unit
Bias	0.01 deg/h ( $1\sigma$ )	6-month recalibration *
Scale-Factor Error	5 ppm ( $1\sigma$ )	6-month recalibration *
Misalignment Coefficients	$5 \times 10^{-5}$ rad	Assumed stable over life of unit
Asymmetry/Nonlinearity	2 ppm ( $1\sigma$ )	Assumed stable over life of unit.

\* Emerging evidence suggests that this recalibration will be accomplished with the unit in the aircraft.



Table 4-6. Navigation-grade accelerometer parameters.

Scale Factor	1000 pps/g	Assumed stable over life of unit
Bias	30 $\mu\text{g}$ ( $1\sigma$ )	6-month recalibration
Scale-Factor Error	50 ppm ( $1\sigma$ )	6-month recalibration
Misalignment Coefficients	$5 \times 10^{-5}$ rad	Assumed stable over life of unit
Cross-Coupling Coefficients	$2 \times 10^{-5}$ rad/g	Assumed stable over life of unit
Scale-Factor Nonlinearity	$<5 \mu\text{g/g}^2$	Assumed stable over life of unit

Because no onboard calibration is assumed, the parameters are assumed to be stable for long periods. The accelerometer stability quoted is readily achievable, while the laser gyros have recently matured to their quoted figures. (The feasibility of using GK analytic-redundancy relationships for laser-gyro and accelerometer recalibration should be explored, as this could lead to a relaxation of the stability requirements with accompanying acquisition cost reduction.)

Regardless of the final number or arrangement of instruments, they will be mounted in prealigned normalized modules with standard interfaces for ease of construction and maintenance, and to enable competitive procurement at all stages of the system life cycle. At this time, four instruments of each type appear sufficient, since a single failure of either type is identifiable using GPS measurements, and the GPS measurements will limit the navigation errors due to an unidentified failure.

The tentative baseline of Table 4-4 includes four display-grade rate gyros (1 degree/hour ( $1\sigma$ )) and accelerometers (10 mg ( $1\sigma$ )) to assure flight-control and display function survivability following a single limited-damage event. These instruments will be mounted separate from the IRA, but their precise arrangement is neither stipulated nor of particular importance. If clustering of all of these instruments is felt to be unwise from a survivability viewpoint, an attractive alternative would be two clusters—each containing two instruments of each type.

The configuration of inertial instruments chosen for the tentative baseline (Configuration 2 in Table 4-3) was chosen over Configuration 1 for two major reasons. First, it has not been demonstrated that the approach of Configuration 1 does indeed assure flight-control function following battle damage. Second, Configuration 1 provides no fault tolerance following a damage event, while the tentative baseline provides single fault tolerance following a damage event using analytic redundancy.

The tentative baseline contains two multipurpose air data probes and three temperature sensors. The multipurpose probes are chosen over a combination of probes and vanes, primarily because of the possible constraints on surface area availability due to the existence of three omnidirectional low-range airspeed systems, but also because of the choice of simple hardware at the price of more complex, yet inexpensive, computer processing. The use of two multipurpose probes will allow single-fault-tolerant operation, with the failed probe isolated using analytic-redundancy relationships, either TD or TK. Because of the low observability of temperature errors in TK or TD residuals, analytic-redundancy FDI is impractical for the temperature sensors, and three instruments are required to give single-fault tolerance. The choice between the Pacer or Rosemount low-range airspeed systems cannot be made at this time. As mentioned earlier, if future testing of the Rosemount sensor indicates that its performance is comparable to the Pacer sensor, it would appear to be the preferable instrument in terms of lower flow interference between sensors and hardware simplicity. The failure-isolation techniques will be analogous for both sensor systems.

The tentative baseline contains only a single GPS receiver and two JTIDS receivers. Failures of the GPS receiver will be detected using GK relationships with IRA outputs. Navigation errors will still be bounded in the absence of an operating GPS receiver as long as one JTIDS receiver is operating. In the tentative baseline, either self-test or GK analytic redundancy will be used for JTIDS failure isolation, with detection in the latter case via direct-redundancy output comparison.

The tentative baseline contains two MLS receivers and three SRLS transceivers. Single-fault tolerance for the MLS receivers will be obtained using GK for fault isolation, while single-fault isolation



for the SRLS transceivers will be obtained via direct redundancy. At this time, it is felt that the shipboard-landing capability will be flight-critical much of the time due to the lack of availability of alternate land-based landing sites. Therefore, single-fault tolerance of the SRLS transceivers throughout the flight appears to be a reasonable requirement.

As mentioned earlier, several interesting research areas have emerged during the course of this study that could impact the generic avionics system baseline design. They are as follows:

- (1) In order to improve the fault coverage of self-test, failure-mode and effects analyses should be performed on all instruments contemplated for the baseline avionics complement. The IISA program could be of particular benefit in performing these analyses on the inertial-instrument candidates.
- (2) The use of embedded microprocessors in instruments or instrument modules provides a significant amount of computational capacity for instrument compensation and self-test. Although the potential benefits of the embedded microprocessor have only recently begun to be explored, the following are present candidates for inertial instruments.
  - (a) Use of interpolation techniques to decrease inherent instrument quantization by at least an order of magnitude.
  - (b) Compensation of instrument parameter and dynamic terms, in particular, temperature and temperature gradient terms—resulting in lower weight and power penalties than precise temperature control.
  - (c) Use of microprocessor-driven logic to determine BITE thresholds.

These and other possible benefits from the use of microprocessors embedded in the avionics instruments should be investigated.

- (3) The survivability and performance of an IRA with half its instruments on either side of a bulkhead should be explored.

- (4) Although geographic kinematics analytic redundancy appears capable of isolating navigation-level failures in inertial-instruments and radio-navigation receivers, the concept has never been applied to systems more complex than idealized point-mass models. The use of GK tests for failure isolation should be thoroughly tested, with particular attention paid to the effects upon performance of uncertainty in the locations of the GPS and JTIDS antennas relative to the IRA and the response of the tests to failure sizes much larger than expected in the flight-control instruments, which could result in vehicle loss if not quickly identified.
- (5) The use of geographic kinematics for inertial-instrument bias calibration, with the aircraft on the ground, should be explored.
- (6) The use of distributed pressure transducers to provide lift and sideforce information at low airspeeds should be investigated. If this technique proves to be a practical source of low-speed air data information for flight control, techniques for assuring reliability of this information source should be explored.
- (7) The Rosemount omnidirectional low-range airspeed system should be thoroughly tested to determine its performance and maintainability.
- (8) The technologies necessary to perform safe, minimum-hovertime landings, aboard ship, in high seas and under all weather conditions should be pursued. A system for both sea-based and land-based platforms should be explored, with high levels of aircraft autonomy and minimal equipment required at the landing site. Ship-motion information must be made available to the aircraft during its approach. Ship-motion prediction is crucial to an automatic system and current efforts in this area should be actively pursued.
- (9) Conflicting opinions exist concerning the ability to reliably isolate individual navigation-level inertial-instrument failures using spatially separated instruments. These conflicts should be resolved under the IISA program for failure levels consistent with future-generation Navy aircraft mission requirements.



To summarize, the interrelationships between functional, fault-tolerance, and survivability requirements and the generic avionics system structure in terms of instrument types and replication have been discussed. Although exact requirements for the avionics system cannot be stipulated at this time, a tentative baseline design has emerged that satisfies reasonable estimates of these requirements. The tentative baseline utilizes analytic redundancy to reduce instrument replication for several instrument types that would otherwise be required to provide fault tolerance. Several research areas have been identified that could have significant impact on the instrument types and numbers in the generic avionics system baseline design.

# LIST OF REFERENCES

- 4-1 Fraser, D.C., et. al., First Quarterly Progress Report, Contract NAS 9-4065, Task Order 41, The Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts, August 1971.
- 4-2 Anon., MIRA, Multi-Function Inertial Reference Assembly, Report MDC A 5329, McDonnell Aircraft Company, St. Louis, Missouri, September 1978.
- 4-3 Weinstein, W., Feasibility and Design Studies of an Integrated Sensor Subsystem (ISS) for Advanced V/STOL Aircraft, Report NADC 76259-30, Naval Air Development Center, Warminster, Pennsylvania, March 1978.
- 4-4 Stonestreet, W. and D. Douglas, "Design of an Integrated GPS/JTIDS/INS System", Presented at the IEEE 1978 Position, Location, and Navigation Symposium (Proceedings of PLANS' 78), San Diego, California, 7-9 November 1978.
- 4-5 Desai, M.N., J.C. Deckert and J.J. Deyst, "Dual Sensor Failure Identification Using Analytic Redundancy," AIAA J. Guidance and Control, Vol. 2, No. 2, March-April 1979.
- 4-6 Anon., NATOPS Flight Manual AV-8A and TAV-8A Aircraft, NAVAIR 01-AV8A-1, Washington, D.C., 1 June 1978, p. 11-13.
- 4-7 Neil, R.D., A State-of-the-Art Assessment of Air-Data Sensors for Naval Aircraft, Master's Thesis, Naval Postgraduate School, Monterey, Canada, September 1977.
- 4-8 Potter, J.E., and J.C., Deckert, "Minimax Failure Detection and Identification in Redundant Gyro and Accelerometer Systems", J. Spacecraft and Rockets, Vol. 10, No. 4, April 1973, pp. 236-243.
- 4-9 Gilmore, J.P. and R.A. McKern, "A Redundant Strapdown Inertial Reference Unit (SIRU)," J. Spacecraft and Rockets, Vol. 9, No. 1, January 1972, pp. 39-47.
- 4-10 Potter, J. E. and M.C. Suman, "Thresholdless Redundancy Management with Arrays of Skewed Instruments", AGARDOGRAPH-224, Integrity in Electronic Flight Control Systems, 1977, pp. 15-25.
- 4-11 Gai., E., J.V. Harrison and K.C. Daly, "Failure Detection and Isolation Performance of Two Redundant Sensor Configurations," Proceedings of PLANS'78, San Diego, California, 7-9 November 1978.



LIST OF REFERENCES (Cont.)

- 4-12 Daly, K.C., E. Gai and J.V. Harrison, "Generalized Likelihood Test for FDI in Redundant Sensor Configurations," AIAA J. Guidance and Control, Vol. 2, No. 1, January - February 1979.
- 4-13 Adams, M., "Impact of IMU Nav Base Errors on IMU FDI" Parts I and II, ISS Memos 75-357 and 76-78, The Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts, December 1975 and March 1976.
- 4-14 Deckert, J.C., Definition of the F-8 DFBW Aircraft Control Sensor Analytic Redundancy Management Algorithm, Report R-1178, The Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts, August 1978.
- 4-15 Wald, A., Sequential Analysis, Dover, New York, 1973, Chapter 3.
- 4-16 Wald, A. and J. Wolfowitz, "Optimal Character of the Sequential Probability Ratio Test", Annals. of Math. Stat. Vol. 19, 1948, pp. 326-339.

## SECTION 5

### DISPLAYS AND CONTROLS

#### 5.1 Introduction

Integrated avionics systems have the potential to expand the operating capabilities of future aircraft, providing great advantages in total-system reliability and maintainability. To complement the increase in aircraft capabilities and mission requirements, the displays and controls must follow similar development. For these cockpit systems, important design goals are: reduced crew workload, unquestioned reliability, and minimum maintenance and support requirements.

In response to this demand for improved displays and controls in vehicles as diverse as helicopters, submarines, spacecraft, and V/STOL aircraft, many prototype systems are under development. The U.S. Navy is currently developing the Advanced Integrated Display System (AIDS) for application to advanced aircraft such as the V/STOL. This section concentrates on how the AIDS can be best integrated with the rest of the avionics system, in order to maximize the advantages of both.

The Navy AIDS and the generic fault-tolerant avionics system are discussed briefly in Sections 5.2 and 5.3 to provide quick reference to the salient features of each. More detailed descriptions are available in References 5-1 and 5-2. Comparison of the capsule descriptions serves to point out the essential structural points in common between the two. Their hierarchial similarities provide the basis for an effective integration plan.

Section 5.4 details the proposed plan for integration of the AIDS system into a generic fault-tolerant avionics system. The integration problem has been approached at two levels, indicating the depth of integration and the amount of change required in combining the two systems.

Level-1 integration is the simplest approach. The AIDS, as configured in the current Advanced Development Model (ADM), is attached to a fault-tolerant network by nodes at the AIDS data-entry points.



Beyond this attachment, the AIDS is completely autonomous. All data and control information flows through the node connection, and the internal AIDS structure is unchanged.

Level-2 integration addresses the integration of an AIDS-concept cockpit into a totally integrated aircraft avionics system. Here, the AIDS ADM is viewed only as a current laboratory implementation of a concept—that of programmable displays and control panels combined with digital data processors. In the Level-2 plan, the functional blocks of the AIDS are preserved, but the hardware locations and communication structure have been altered to take advantage of a fault-tolerant architecture, redundancy management, and expected changes in hardware capabilities.

In Level-2 integration, particular care is paid to the preservation of flight-critical information and display elements, as opposed to that which is only mission-critical. Dividing display requirements and data flow on the flight/mission-critical line dictated the restructuring of internal AIDS networks, and the use of fault-tolerant computation. This integration approach provides the flexibility intended in the AIDS concept, combined with the reliability that comes from full integration.

Section 5.5 (Conclusions and Recommendations), contains a discussion of the directions towards which future AIDS developments might proceed, in order to enhance integration with a total aircraft avionics system.

## 5.2 Advanced Integrated Display System (AIDS)

The Navy AIDS is a program under study by the Naval Air Development Center (NADC) and General Electric to develop a modular, flexible, and programmable integrated display and control system for future aircraft. Currently in the advanced-development-model stage, wherein new technologies are experimentally examined and hardware developed, the AIDS will ultimately serve as a base for engineering development efforts on specific weapon systems. Figures 5-1 and 5-2 depict a current cockpit display layout and hardware configuration for the AIDS ADM.

The technical approach in the AIDS has been to develop a minimum number of simple display units (cathode-ray tubes (CRTs)) and control units (dedicated and programmable keyboards) on three bus systems: a digital bus, a video bus, and a power bus. Control- and signal-processing

GENERAL ELECTRIC AIDS COCKPIT CONFIGURATION

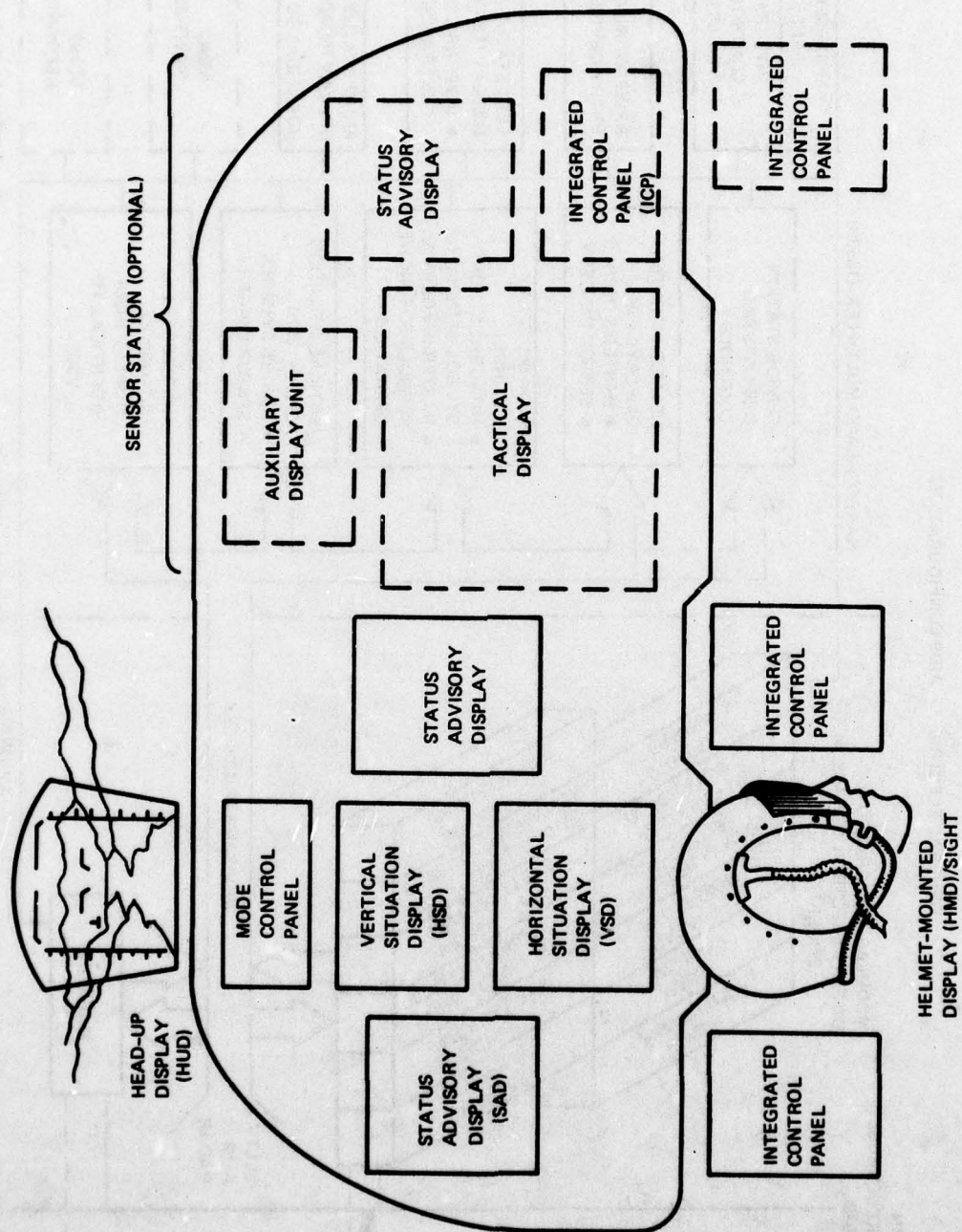


Figure 5-1. AIDS cockpit configuration.



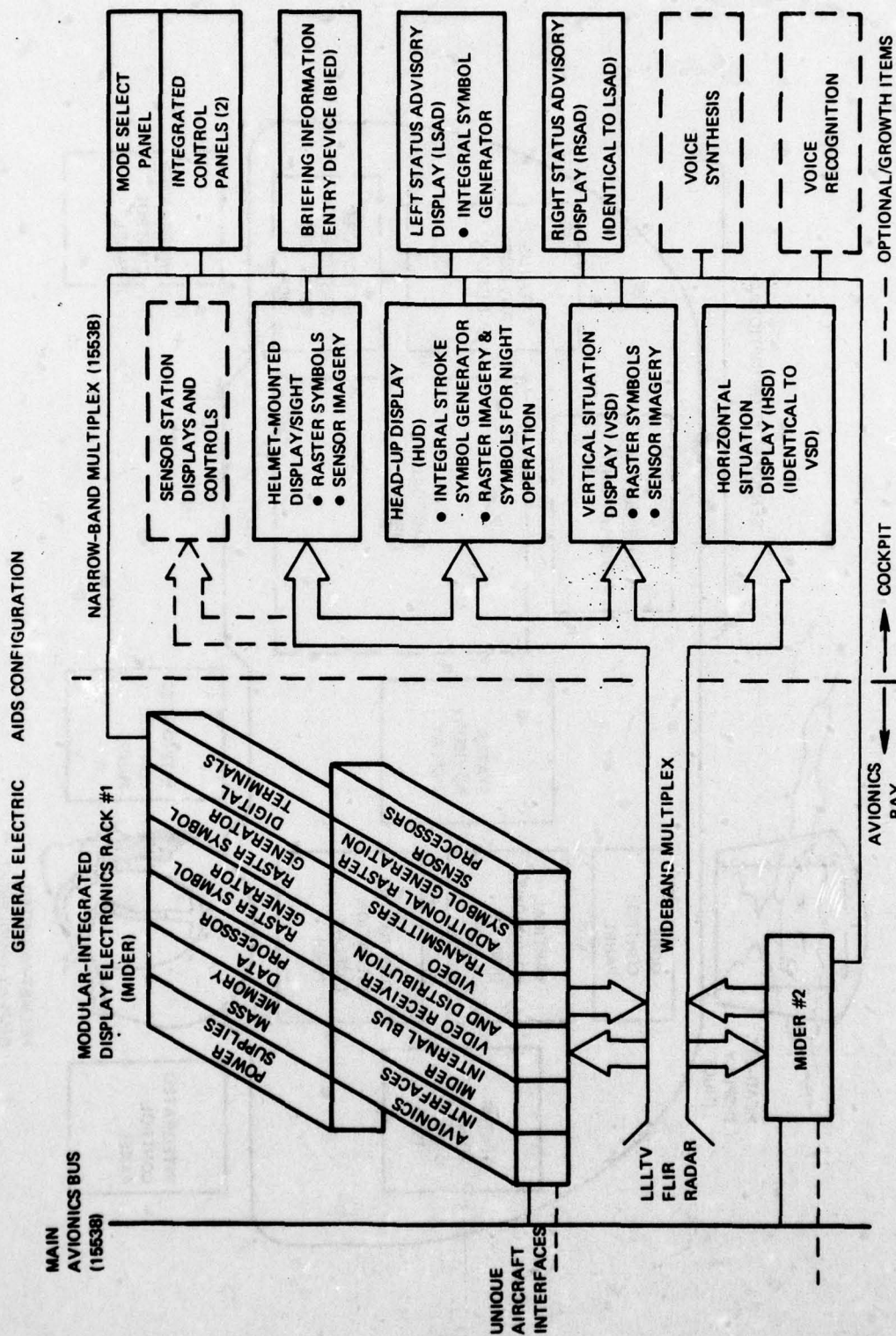


Figure 5-2. AIDS system configuration.

hardware, made up of modular programmable units, drives the display/control units via the buses in response to manual and automatic command signals. The AIDS central processor is, in turn, in communication with external aircraft systems, and provides for data flow into AIDS and pilot commands out to other subsystems.

Detailed descriptions of AIDS hardware and functioning are contained in References 5-1 and 5-2. For purposes of clarity in further discussion of integration, brief descriptions of major items are included herein. Figure 5-2 shows the AIDS ADM, partitioned into: (1) cockpit displays and control panels, (2) the modular integrated display electronics racks (MIDER), and (3) the external aircraft avionics system.

The purpose of the MIDER is to house the equipment that controls and conditions inputs from the outside subsystems and generates display symbology. In addition, under operator control, it selects and conditions outputs to the external subsystem.

Inputs arrive from the avionics system, but via a 1553B terminal. This information, entering MIDER #1 or #2, is transmitted on an internal MIDER bus to any element requiring the data. Key elements in the MIDER are its main data processor, the mass memory, video-receiver-transmitter equipment, and raster signal generators (RSGs).

The data processor performs all functions associated with overall system control, built-in-test, mode selection, display formatting, and configuration. Control signals are transmitted to the cockpit units via a 1553B bus.

The nonvolatile mass memory is used for storage of display formats and operating programs.

Video receiver/transmitter equipment is used to ensure synchronization between external video inputs and AIDS symbology, so that both can be superimposed on a display screen.

The function of each RSG is to generate in-raster symbols for an associated display. The output of the RSG is a video signal, which is either mixed with external video or sent directly to the display via the wideband multiplex bus.



### 5.3 Fault-Tolerant System Network and Architecture

The purpose of this section is to provide a brief summary of the hierarchical fault-tolerant network and architecture. Although this discussion is brief and somewhat repetitive of Section 3, it is included herein to illustrate the compatibility between the AIDS system and the fault-tolerant architecture. Further information is contained in Section 3 and in References 5-3 and 5-4.

The fault-tolerant system has a very high level of integration of all avionic functions, with flexible communication paths throughout. Information generated anywhere within the system can be made available anywhere else in it. Alternate means for fulfilling various data needs make survival possible in the face of failures or damage.

Specifically, the integrated system consists of the following:

- (1) One or more high-level fault-tolerant multiprocessors (FTMP).
- (2) Some number of local processors, each of which bears a unique relationship with one or a small group of subsystems. Such computers may be physically embedded in the related subsystem.
- (3) A possible intermediate level of computers dedicated to certain tasks.
- (4) A communications network to which all computational sites are attached—each computer being attached at one or more nodes of the network.

High-level or critical functions are performed by highly reliable fault-tolerant multiprocessors. These multiprocessors, reflecting state-of-the-art, high-performance microprocessor technology, will have substantial computational power—able to perform critical flight control, autoland, and thrust control, plus system configuration and restructuring functions.

Lower level positions in the hierarchy are generally serviced by dedicated microprocessors. Most of these functions are associated with equipment that has a failure rate considerably greater than a simplex microprocessor's. Loss of a dedicated local processor is equivalent to loss of the associated subsystem, component, or sensor. However, the failure rate of the combined dedicated simplex processor and subsystem or component is not significantly greater than that of the subsystem or component alone. Figure 5-3 illustrates the hierarchical structure.

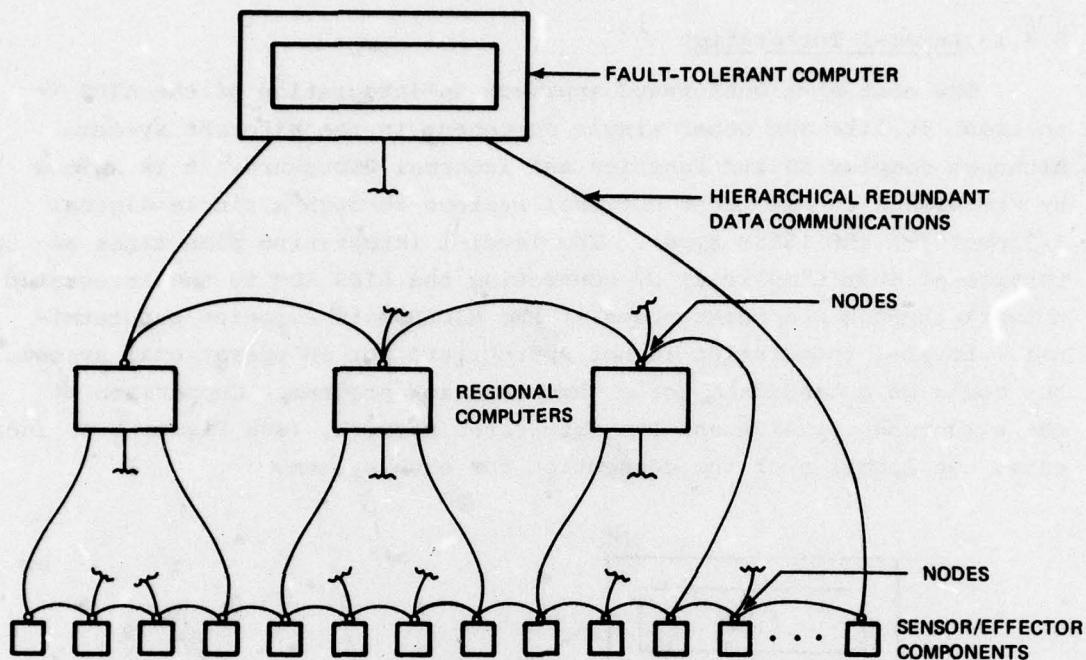


Figure 5-3. Fault-tolerant system hierarchy.

Computational sites are joined by a network of links—a link being a fully duplex communication path between any two computational sites, or nodes. Each node is interconnected to at least two other nodes. Each node also contains switching circuitry so that the links can be connected. Thus, if a link is viewed as an I/O bus segment, a node can, by making the appropriate internal switch-closures, extend a bus through itself or cause it to "Y". The lead node can build an I/O bus, which reaches all other nodes by issuing commands that cause other nodes to set up a branching bus structure. Not all links are used in this process; some remain idle. In the event of physical damage or node failure, the lead node identifies the failure and bypasses the failed or damaged units by activating idle links and nodes.

#### 5.4 AIDS Integration With Core Avionics

This subsection addresses the incorporation of the AIDS architecture into the overall integrated avionics architecture. Two approaches to this incorporation are presented.



#### 5.4.1 Level-1 Integration

The most straightforward approach to integration of the AIDS is to treat it like any other single component in the aircraft system. Although complex in its function and internal structure, it is suited by its design to mate with external systems through a single digital I/O port (of the 1553B type). The level-1 integration plan takes advantage of this simplicity by connecting the AIDS ADM to the integrated network through redundant nodes at the MIDER main avionics bus terminal. Level-1 integration is not appropriate for an operational system, but could be a candidate for a demonstration program. Comparison of the structures of AIDS and the integrated network, (see Figure 5-4) indicates the location of the connection for each system.

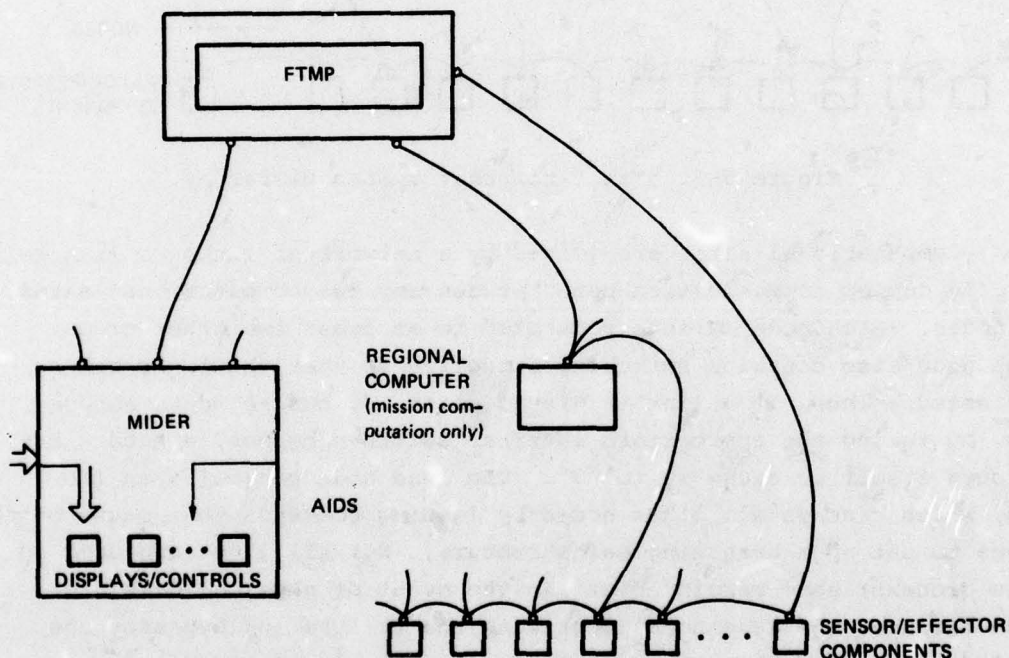


Figure 5-4. Level-1 integration of AIDS.

Figure 5-4 illustrates the reduction of the entire AIDS to a single functional block, with its connection to the network. Since all of the data processing for the displays and controls is handled internally, the AIDS places small data-rate requirements on the network. The FTMP serves in a status monitoring capacity, and provides updated state information to the AIDS processor. The middle level or regional computer is not required for flight-critical display and control calculations, which must be in the FTMP, but it may be necessary for the mission-related computation. Further study and specific mission requirements will be necessary to determine the need for such a regional processor.

The Level-1 approach has several benefits that arise directly from its simplicity. Since the only requirement on connection is that of compatible data formatting, the currently available AIDS ADM hardware can be readily combined with a prototype network. This combination can provide a test bed for further research and human-factors studies. With suitable dedicated backup equipment, the combined system could also be used in a flight demonstrator for proof-of-concept tests.

For operational use, however, the Level-1 integration plan does not take full advantage of the fault tolerance available with more extensive integration. The core avionics should have a probability of causing a vehicle loss of about  $10^{-7}$  per hour. As a first attempt at partitioning this probability, it is assumed that the display and control system should have a probability of failure that can cause a vehicle loss of less than  $10^{-8}$  per hour. As will be explained in the following paragraphs, a dual system (such as the AIDS ADM configuration) cannot achieve this level of fault tolerance. The Navy recognizes this problem and plans to incorporate dedicated backup displays so that AIDS failures will not cause vehicle loss. Section 5.4.2 discusses a Level-2 system using AIDS elements, which is intended to have adequate fault tolerance so as not to require dedicated displays.

The following simple analysis shows that a dual system is not likely to achieve the level of reliability necessary to support flight-critical functions. There are two ways that a dual system failure can cause a vehicle loss. They are as follows:

- (1) A single undetected failure, which causes loss of display without automatic reconfiguration or without time or information for the pilot to reconfigure the system.
- (2) Dual failures for components that are only dual redundant.



The probability of a loss of flight-critical displays from these failures can be written approximately as:

$$P_L = (1 - c)P_f + P_f^2 \quad (5-1)$$

Where  $P_L$  is the probability of a loss of flight-critical displays,  $c$  is the coverage for the first failure (defined as the probability of detecting and successfully reconfiguring after that failure), and  $P_f$  is the probability of a single failure.

It is unlikely that either term on the right of Eq. (5-1) can be kept below  $10^{-8}$ . The first term involves the coverage  $c$ . To obtain coverage, the AIDS uses BITE, and the best that can be expected for BITE is about 95-percent coverage. For the first term to be less than  $10^{-8}$ , then the MTBF of any critical component (the loss of which can only be detected by BITE) would have to be greater than  $5 \times 10^{-6}$  hours, assuming a 1-hour mission. This is much better than can be expected. It can be argued that the crew can detect faults to a much higher level than BITE; however, subtle failures may not be detectable at the level of coverage required.

The second term corresponds to two failures than can fail the AIDS. For this term to be less than  $10^{-8}$ , the component MTBFs must be:

$$MTBF > \sqrt{10^8} = 10^4 \text{ hours}$$

This is still unrealistic, so that even perfect BITE is not likely to solve the fault-tolerance problem for this configuration.

General Electric has performed reliability studies of the AIDS\* showing that, as we understand the data, the probability of vehicle loss is below  $0.2 \times 10^{-6}$  per mission. This probability is better than that indicated by the simplified analysis already mentioned, partially due to the fact that the AIDS ADM is more than dual redundant. However, the General Electric analysis does not seem to include the possibility of undetected failures, and so seems to be optimistic.

#### 5.4.2 Level-2 Integration

Level-2 integration is a plan for incorporation of the concept of programmable displays and control panels, (currently embodied by

---

\* See page 311 of Reference 5-1.

AD-A065 136

CHARLES STARK DRAPER LAB INC CAMBRIDGE MA

F/G 1/3

AN INTEGRATED FAULT-TOLERANT AVIONICS SYSTEM CONCEPT FOR ADVANC--ETC(U)

FEB 79

N00019-78-C-0572

UNCLASSIFIED

R-1226

NL

2 OF 4

AD  
A065136





the AIDS ADM hardware) into the integrated avionics network. The functional blocks in the AIDS, as shown in Figure 5-5, are present in the Level-2 integration, although their locations and communication links have been changed to utilize the fault-tolerant hierarchy. As was the case with the Level-1 integration, the regional computer is not required for flight-critical calculations, which must be performed in the FTMP, but it may be necessary for mission-related computations. Since the displays and controls are flight-critical elements of the avionics system, any integration plan must give primary emphasis to ensuring reliability of these units. All elements of the cockpit systems (data processing, buses, signal-processing hardware, and CRTs) must be configured with this in mind.

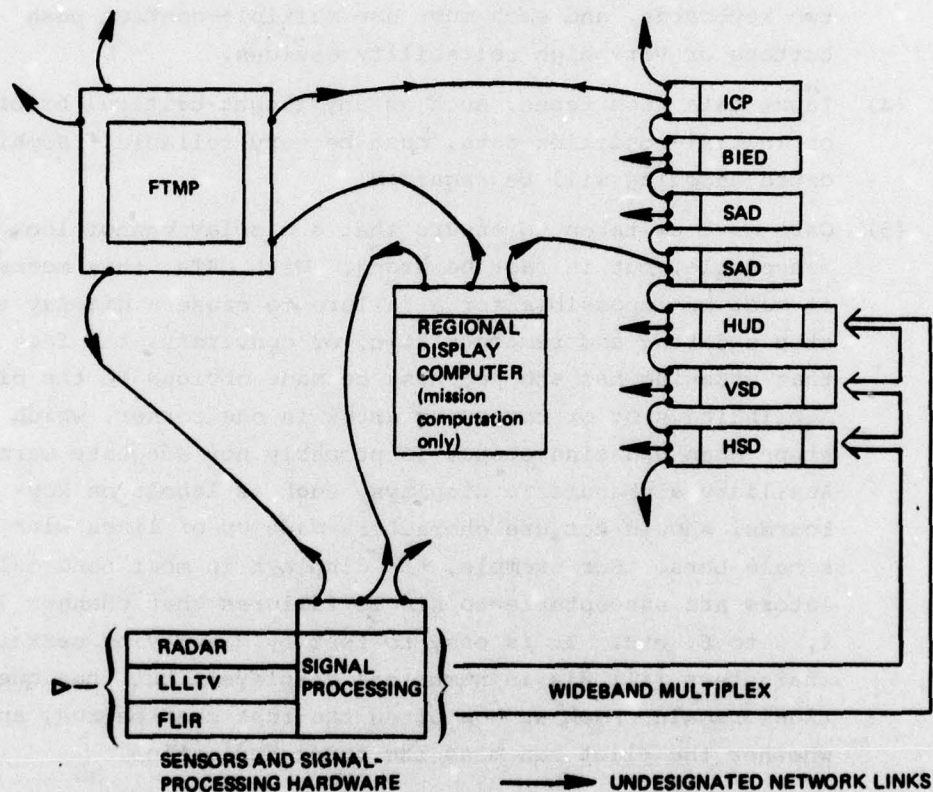


Figure 5-5. Level-2 integration of AIDS.

Some specific details to be considered are:

- (1) The hardware configuration must have enough redundancy to support two failures of elements involved in flight-critical functions. This may involve triple redundancy, and/or

system reconfiguration. For example, it must be possible to display all flight-critical information on either the horizontal situation display (HSD), vertical situation display (VSD) or head-up display (HUD) \*. In other words all three devices would have to fail to endanger the vehicle. The pilots ability to fly from any display needs study. This is a possible future question for Design Evaluation Flight Test (DEFT)\*\*.

- (2) Coverage for at least the first failure must be very nearly 100 percent. Specific critical areas are included in the following items.
- (3) Redundancy and fault detection must be built into keyboard devices. It must be possible to enter data from at least two keyboards, and each must use multiple-contact push buttons or very high reliability devices.
- (4) Input data from tapes, such as any flight-critical briefing or initial-condition data, must be very reliable. Sophisticated encoding will be required.
- (5) Care must be taken to ensure that a display cannot look reasonable, but in fact be wrong. With CRTs, this means it must be impossible for a failure to cause a display to stop updating and remain static, or conversly, the fact that updating has stopped must be made obvious to the pilot. A blinking spot or reversing arrow in one corner, which stops when updating stops, is probably not adequate warning. Auxiliary alphanumeric displays, such as labels on keyboards, should not use characters made up of lines with single bars. For example, the displays in most hand calculators are susceptible to single failures that changes 7 to 1, 8 to 6, etc. It is easy to test by displaying certain characters (all 8's in numerical displays), but then questions remain, such as how often the test must be run, and whether the pilot can miss the error indication.

---

\* It is generally assumed that the display-generating part of a HUD (e.g., CRT) cannot be redundant, and so cannot be of very high reliability. However, Reference 5-5 describes the possibility of redundant display elements. This should be explored further.

\*\* See Reference 5-6 for more details.



If the aforementioned requirements are met, it becomes reasonable to require very high fault tolerance from the integrated display system. A detailed analysis should be performed as the design progresses, to ensure that adequate reliability is achieved.

The FTMP and its network of nodes should be incorporated into the display system structure. This adds the strengths of the multi-processor and the redundancy-management system to the flexibility of the programmable displays. The block diagram in Figure 5-5 indicates the Level-2 plan. When compared to the diagram of the AIDS ADM (see Figure 5-2), it can be seen that the functions carried out in the MIDER have been moved up or down in system hierarchy. As configured by this plan, the data-processing, system-control, and management functions have become the duties of the FTMP; or they have been assigned to a regional computation site. Each of the display or control elements has become a separate node on the network, joined by a highly reliable, flexible network. The raster-signal generators and video hardware formerly in the MIDER have been combined with the CRT electronics of each display. Depending on the level of memory technology (and costs), the format memory unit may be in the FTMP or duplicated at each display.

In this organization, the FTMP and its reliable network are used to prepare and transmit the flight-critical elements of information to and from the selected nodes.\* This flight-critical information includes all alphanumeric data (selected by the pilot for display during a particular flight phase), and the command signals sent by the control panels and input devices. The mission-critical information (i.e., video signals from RADAR, LLLTV, or FLIR) is external to the fault-tolerant network.

The mission-critical video signals from aircraft sensors can be provided to the selected display (HSD or VSD) by a separate video bus. The video signals are not compatible with the fault-tolerant network, but since they are not flight-critical, they don't need its benefits in higher reliability. The network can, however, provide management of the video bus. This division of signals provides most protection where it is most needed. The video and alphanumeric data can be combined at the selected display; a function carried out in the AIDS

---

\* Any one of the five CRTs (HUD, VSD, HSD, SAD x 2) could be used for critical symbology, rather than just the primary three.

MIDER. By delaying this mixing until the data reach the display, the flight-critical and mission-critical information travels over paths of appropriate bandwidth and reliability.

In this Level-2 organization, the individual display unit assumes more of the overall signal-processing responsibility than in the AIDS ADM. In addition to its node connection and control electronics, each display must have a RSG memory, and possibly video-mixer capability. Since the technological trend in analog and digital circuit design is toward ever larger scale integration, it is not inconsistent to expect that greater capabilities can be built into dispersed hardware sites.

The rate at which the AIDS will require data from the network can not yet be firmly established. However, an estimate is available from Table 35 of Reference 5-5. This table lists data requirements from 2.9 to 38 percent per megabit/second available on a 1553 bus. The 2.9 percent is for a single-seat fighter, and the 38 percent is for ASW search and classification. If compatibility for ASW were to be part of the design, a provision might have to be made for dedicating a bus to this function when required.

#### 5.5 Conclusions and Recommendations

The Navy AIDS is compatible with an integrated avionics design, and can be readily integrated into the system in either of two configurations. Most directly, the AIDS ADM can be attached to nodes of the integrated system. This takes maximum advantage of currently available hardware, but does not provide the fault-tolerant potential required for future aircraft applications. A second configuration redistributes the functional blocks of the AIDS within a fault-tolerant hierarchy, separating data paths for flight- and mission-critical information. This system retains the AIDS design goals of flexibility, modularity, and programmability, while also offering requisite levels of fault tolerance. The following areas are recommended for further effort.

- (1) Human-factors studies, as embodied in the DEFT program, should increase. Baseline display formats need to be selected, in order to more firmly establish the memory size and data rates required in an operational system. Inherent flexibility will tolerate the natural evolution, but initial formats should be established. Further questions, such as the identification of minimum display requirements for



flight and landing, and the increase in workload due to loss of a particular CRT (i.e., HUD OR HSD), should be addressed.

- (2) A program to combine various AIDS ADM hardware units with an integrated avionics system should be initiated. This would provide a flight-test-bed demonstration, as well as a laboratory "hot mockup".
- (3) A detailed study is needed of the benefits and costs of allocating redundant analog and digital hardware to the individual CRT. This would include evaluation of failure probabilities, life-cycle costs, and expected technical capabilities for the 1990 time frame.
- (4) A set of interface specifications should be determined to ensure compatibility between diverse development efforts.

#### LIST OF REFERENCES

- 5-1 Advanced Integrated Display System (AIDS) System Design Interim Report No. 3 Vol. I, General Electric, Aircraft Equipment Division, Utica, New York 13503, 31 October 1977.
- 5-2 Advanced Development Program Plan, Advanced Integrated Display System (AIDS) Project W0597 A supporting Technology for Type A V/STOL, NADC, 5 December 1977.
- 5-3 Type A V/STOL Avionics Vol. 2 Technical Proposal, The Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts, June 1978.
- 5-4 J. J. Deyst and A. L. Hopkins, "Highly Survivable Integration Avionics", Aeronautics and Astronautics, September 1978.
- 5-5 Head-up Displays: A literature Review and Analysis with an Anotated Bibliography FAA NA-77-42 Jack J. Shrager National Aviation Facilities Experimental Center, April 1978.
- 5-6 D., Eliassen, Lt. Col. USAF, and H. Levin, Design Evaluation Flight Test (DEFT), 51-751 Air Crew Symposium NATC, 10 May 1978.

## SECTION 6

### RADIO NAVIGATION AND COMMUNICATIONS ALTERNATIVES/REQUIREMENTS

#### 6.1 Introduction

This section discusses approaches to the radio-navigation and communication aspects of the avionics system. The radio-navigation and communication systems discussed are those that are most likely to meet the core avionics requirements of tactical aircraft in the late 1990s. Some of these systems are currently operational (e.g., Tactical Air Navigation (TACAN) and Identify Friend or Foe (IFF)), while others are still in their development phases (e.g., NAVSTAR Global Positioning System (GPS) and Joint Tactical Information Distribution System (JTIDS)). This section is primarily concerned with the transceiver and signal and data-processing functions. The control/display (C/D) functions are discussed in Section 5.

Two levels of integration are considered; in Level 1, the outputs from separate pieces of equipment are integrated; in Level 2, the separate equipments are also integrated—specifically, the concepts of the Tactical Information Exchange System (TIES) are employed in this integration. In either case, the outputs of these systems are integrated in a manner to attain required performance while at the same time attempting to maximize fault tolerance.

In the following sections each of the systems are described, then alternative configurations employing both approaches are discussed.

#### 6.2 NAVSTAR Global Positioning System (GPS) (6-1, 6-2, 6-3)

##### 6.2.1 System Description

NAVSTAR GPS is a spaced-based radio-positioning navigation system that will provide extremely accurate three-dimensional position and velocity information together with system time to suitably equipped users anywhere on or near the earth. GPS consists of three major segments: Space Segment, Control Segment, and User Segment (see Figure 6-1).



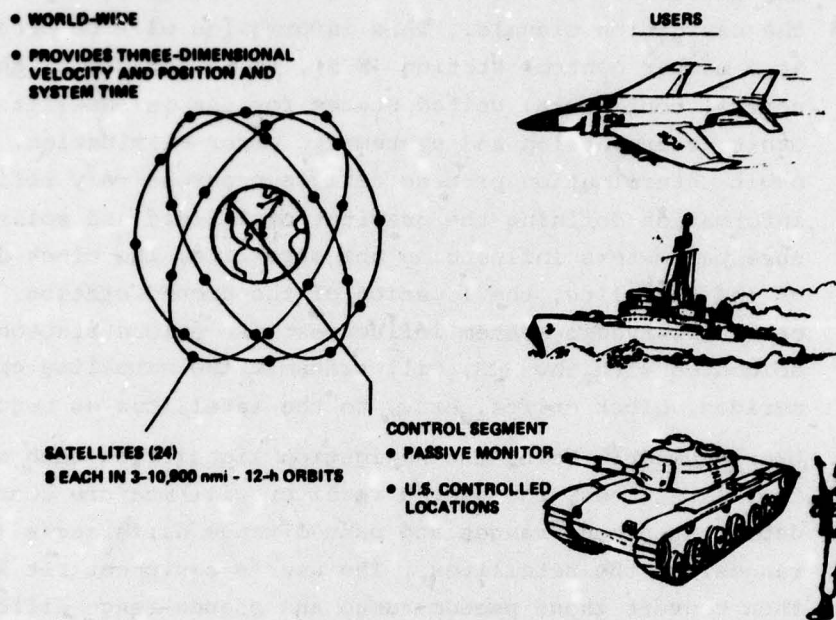


Figure 6-1. NAVSTAR GPS—system concept.

- (1) Space Segment: The fully operational GPS will deploy 24 satellites in three groups of eight circular, 10,900 nautical-mile orbits inclined at approximately 63 degrees, and having a 12-hour period. This deployment will provide the satellite coverage for continuous, three-dimensional position and velocity determination. Each satellite will transmit L1 and L2 composite signals at 1575.42 and 1227.6 MHz consisting of a precision pseudo-random noise (PRN) navigation signal and coarse acquisition PRN navigation signal. The signals contain biphase-modulated navigation data such as satellite ephemeris, and satellite-clock correction information. Use of both the L1 and L2 signals permits the user to determine the ionospheric-group delay or other electromagnetic disturbances in the atmosphere, which may affect the transmitted signals.

- (2) Control Segment: Five widely separated monitor stations (MS), located on U.S. controlled territory, will passively track all satellites in view, and accumulate ranging data from the navigation signals. This information will be processed at a master control station (MCS), to be located in the central continental United States for use in satellite-orbit determination and systematic error elimination. The orbit-determination process derives progressively refined information defining the gravitational field and solar pressure parameters influencing the satellite, the clock drift of the satellite, the location of the ground station, and other observable system influences. An upload station, colocated with the MCS, will transmit the satellite ephemerides, clock drifts, etc., to the satellites as required.
- (3) User Segment: Using the navigation signal from each of four satellites, the user's receiver will measure four independent pseudo-ranges and pseudo-range differences (delta ranges) to the satellites. The user's equipment set will then convert these pseudo-range and pseudo-range differences to three-dimensional position and velocity and system time. This position solution is in World Geodetic System Coordinates—an earth-centered earth-fixed coordinate system, which can be converted to any coordinate frame in units of measure required by the user.

#### 6.2.2 User Equipment Description

The purpose of the GPS baseline set, which is part of the user segment of GPS, is to receive the signals transmitted by the GPS satellites and process them to provide highly-precise three-dimensional position and velocity and system-time information. Each satellite will transmit two distinct PRN-modulated radio frequency (RF) signals at L-band; a precision (P) navigation signal (10.23 M chips/second), and a coarse/acquisition (C/A) navigation signal (1.023 M chips/second) at the L1 frequency (1.57542 GHz); and either the P signal or the C/A signal at the L2 frequency (1.2276 GHz).

The GPS baseline set is essentially the X-set designed by the Magnavox Government and Industrial Electronics Company, Advanced Products Division, Torrance, CA. (6-4)



A functional block diagram of the GPS X-set is shown in Figure 6-2. The set consists of two antennas, two preamplifiers, a receiver, a signal processor (process controller), a data processor, and a power supply. Each of the two antenna receives signals at both the L1 and L2 frequencies. The preamplifiers raise the input signal level, thus establishing the input noise figure. The receiver, under control of the signal processor, acquires the satellite signals, tracks the carriers and the codes (either the P or C/A), demodulates the incoming data, and measures the psuedo-range, delta-range, and ionospheric propagation delay. The data processor selects the satellites to be tracked, and performs the calculations to provide the navigation data.

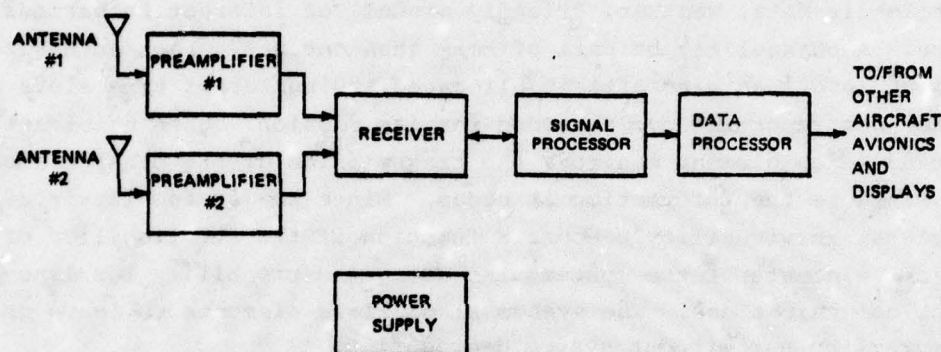


Figure 6-2. Functional block diagram of GPS X-set.

The GPS X-set has the capability of using an internal-reference oscillator or an external-reference oscillator as a frequency source, and/or, an external clock for accurate time-of-week information. This set is also capable of using data from an inertial measurement unit (IMU) in the navigation filter to provide improved velocity and position estimates.

Detailed descriptions of each part of the GPS X-set are given in Appendix 6-A.

### 6.3 Joint Tactical Information Distribution System (JTIDS)

#### 6.3.1 System Description

JTIDS is a secure, jam-resistant, digital information-distribution system with relative navigation, and positive user-identification capabilities that will be suitable for use by all services. <sup>(6-5)</sup> JTIDS is planned to be used within a mix of alternative communications resources to interconnect the tactical and air defense elements of all services, including surface and airborne command/control, surveillance and intelligence centers, ships, and combat and support aircraft.

Precise system and signal time-of-arrival measurements, coupled with the transmission of emitter location, permit users to position themselves within an established two-dimensional/relative-navigation grid. In Phase II the system will use an advanced time division multiple access (TDMA) technique to interconnect all system users into a single net for simultaneous distribution and reception of information. A net is that collection of all time slots and user interconnections within a recurring block of time known as an epoch. All net participants must use a common PRN code and a common time. A typical net is shown in Figure 6-3. A net can be a collection of channels which are sets of recurring time slots assigned to specific functions (e.g., hostile-air data, weather, friendly ground) of interest to various users. A channel may be part of more than one net. Each authorized element (e.g., an aircraft) is allocated the number of time slots within the net reporting cycle needed for its mission. When not transmitting, each element monitors the transmission of the other elements and extracts the information it needs. Since the system is virtually nodeless, survivability becomes a function of the survivability of the various elements of the system, including the capability for line-of-sight communications. The system also allows elements to leave or enter the net without system degradation.

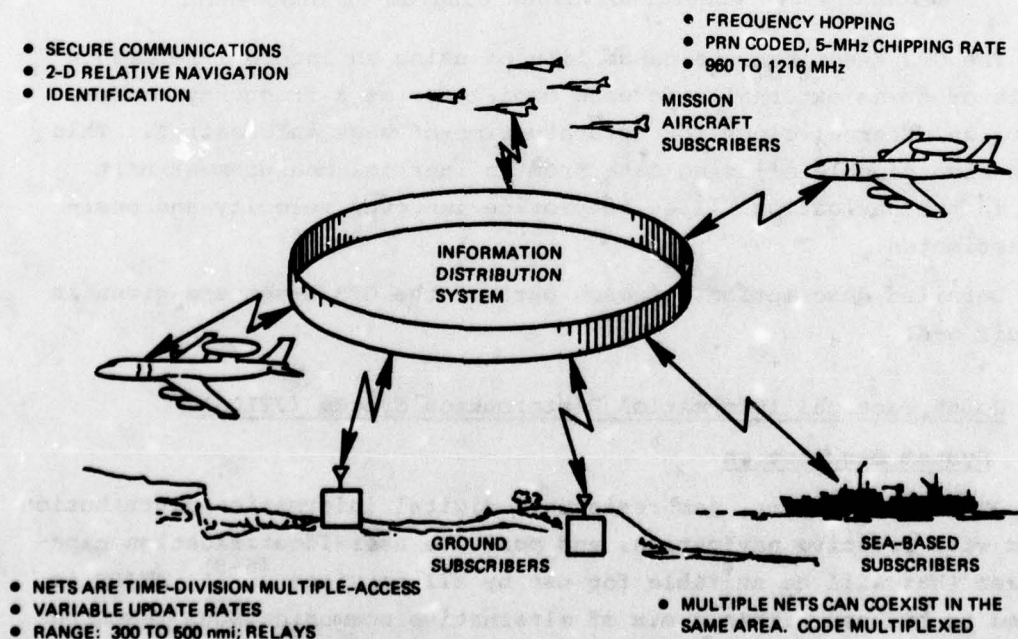


Figure 6-3. JTIDS net.



The employment of spread-spectrum and frequency-hopping techniques provide for electronic counter measures (ECM) protection. With these techniques, information is transmitted over a frequency bandwidth several-thousand times the bandwidth needed to support the transmission of the actual information. The bandwidth expansion is performed at the transmitter. Contraction to the information bandwidth is accomplished by reversing the process. This technique forces the enemy to spread his jamming energy over several-thousand times the bandwidth otherwise needed, resulting in the dilution of his effective energy. Connection of users who are beyond line of sight of one another is accomplished through the use of a relay. It is expected that aircraft would be used for this purpose. Any aircraft already having a JTIDS terminal can be used as a relay. Additionally, relay units can be pod or pallet mounted for easily installation in vehicles not otherwise needing terminal equipment. Figure 6-4 shows some examples of JTIDS applications.

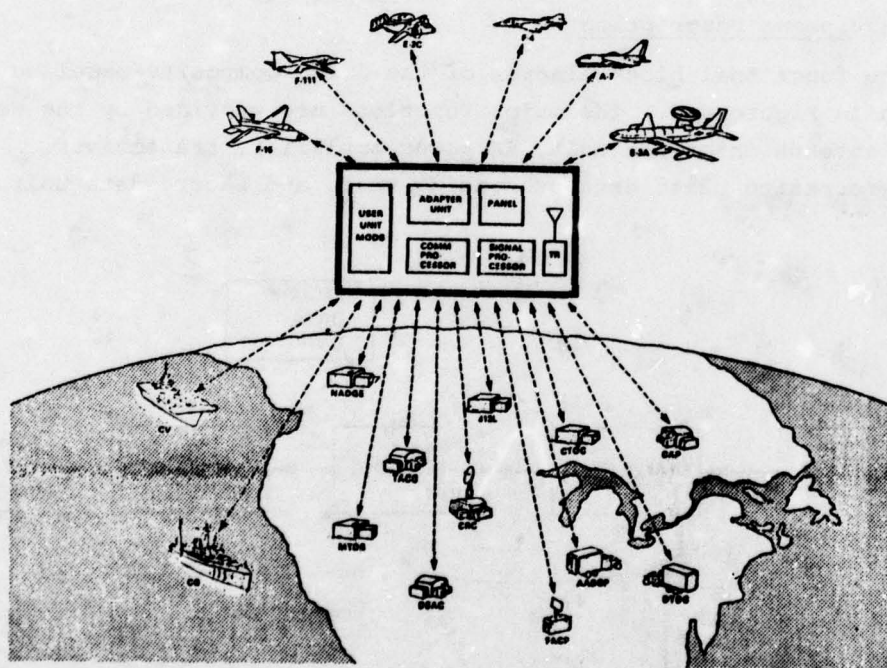


Figure 6-4. Potential JTIDS participants.

Security is an integral part of the JTIDS design. Cryptographic security is employed to eliminate the possibility of enemy eavesdropping, spoofing, or exploitation.

There are several classes of equipment involved in the joint development program. Class I, the initial-design full-capability terminal based on available components and established design concepts, is aimed at early Air Force implementation with E-3A Aircraft. The Class-I terminal will be used in large aircraft, some classes of ships, and in a variety of command-and-control center operations. Class-II terminals will be smaller and engineered for combat aircraft and installation where space and weight are at a premium. This is the terminal of interest in this investigation. (6-6, 6-7) A third class to be developed is a miniterminal which would have applicability for air-traffic control, missile-guidance control, and manpacks. A fourth effort is the Adaptable Surface Interface Terminal, which can provide interface between the JTIDS system and existing C<sup>2</sup> systems.

#### 6.3.2 Equipment Description

The functional block diagram of the JTIDS composite-baseline set is shown in Figure 6-5. The major functions are provided by the antennas, antenna interface unit, RF power amplifier, transceiver, signal-processing unit, data-processing unit, and secure-data unit.

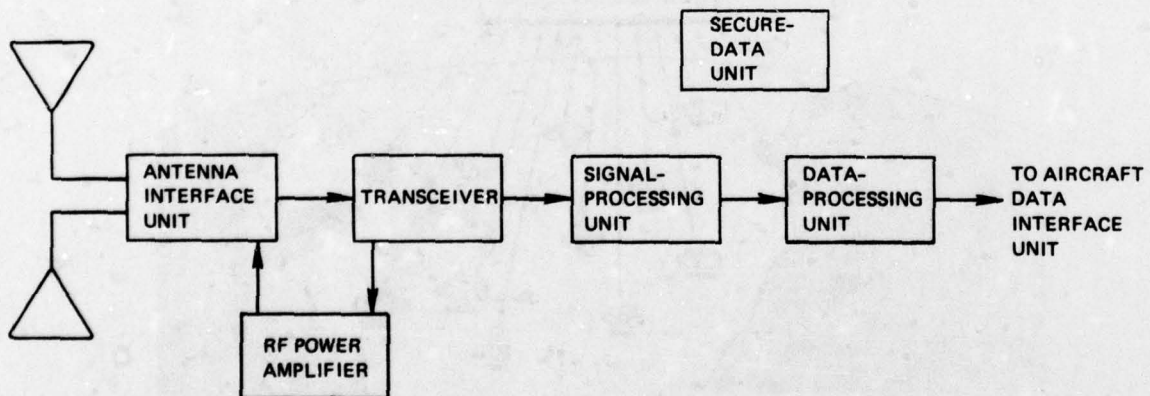


Figure 6-5. Functional block diagram—JTIDS Class-II terminal.



The transmitted signal is routed from the transceiver through the RF power amplifier and antenna interface unit to the antennas. During transmission, cyclic code-shift keyed (CCSK) data is minimum phase-shift keyed (MPSK) onto a frequency-hopped local oscillator (LO), and then up-converted using the same LO signals as the receiver channels. The receiver signal from the antenna is down converted twice with a fixed first LO and a frequency-hopped second LO. Eight parallel receiver channels are used for preamble detection, and one channel is used as a data channel. The second LO frequency for each channel is developed from one of the eight synthesizers which are controlled from the signal processor, as previously described for transmission.

The signal-processing unit and data-processing unit perform the basic message formatting and terminal synchronization. The secure-data unit works with the two processors for data encryption and decryption.

During transmission and reception, the interface between the analog (RF/IF) and digital subsystems is via the signal-processing unit with digital data routed through the secure-data unit. The data-processing unit provides the I/O interfaces and the I/O multiplexer bus for interfacing with other auxiliary or peripheral devices and the central computer on the aircraft.

The data-processing unit performs several other important functions, including: (1) coordinate conversion of received position data, (2) interfacing with the signal processor and units outside the terminal, (3) control of net processing and time synchronization, (4) operator interface for the control and display panel, and (5) message reformatting.

Appendix 6-B describes these functions in greater detail.

#### 6.4 Tactical Air Navigation (TACAN)

##### 6.4.1 System Description

TACAN is a method of providing area navigation for military aircraft. (6-8) It consists of two major components, an aircraft interrogator and a ground transponder or beacon. The aircraft interrogator measures both distance and bearing to the ground transponder. It operates in the 960 to 1215 MHz frequency band. The ground transponder consists of a constant-duty cycle distance measurement beacon. Rotating parasitic elements are added to the beacon antenna to provide

an amplitude modulation (AM) to the transmitted signal. The aircraft interrogator transmits pulses on one of the many frequencies spaced 1 MHz apart in the aforementioned frequency band. (The pulses are transmitted in pairs in order to minimize interference from other pulsed systems.) The ground beacon receives these pulses, and after a fixed delay transmits them back to the aircraft on a different frequency. By measuring the elapsed time between transmission and reception of the pulses at the aircraft interrogator, the distance to the transponder can be determined. By demodulation of the amplitude of these pulses, bearing to the transponder can be determined. The transponder can also be onboard a ship or aircraft.

#### 6.4.2 Equipment Description

The equipment set being developed by the JTIDS Joint Program Office (JPO) has the capability of processing TACAN signals. Thus, there is no separate equipment set recommended for TACAN. See Section 6.3.2 and Appendix 6-B for a functional description of this equipment.

### 6.5 Identify Friend or Foe (IFF)

#### 6.5.1 System Description

The IFF system is used to identify vehicle (generally airborne) status prior to the vehicle being within visual view. It consists of two units, a transponder unit (onboard friendly vehicles) and an interrogation unit. The transponder receives coded interrogation radio signals which originate at a ground, shipboard, or airborne IFF/ATC station. The interrogation signals are detected, decoded, and used to automatically actuate the transmission of a coded reply signal. The interrogating IFF/ATC station decodes the replies to provide identification, altitude, and position information. The interrogator transmits at 1030 MHz, and the transponder transmits at 1090 MHz. The transponder is the unit of interest in this investigation.

#### 6.5.2 Equipment Description

This equipment set would be similar to those currently being used for IFF. An example of this equipment is the APX-100. The APX-100 is different from most IFF sets in that the signal-processing electronics are located with the control display unit (CDU) in the cockpit. This would not necessarily be the best configuration for future tactical aircraft.



## 6.6 UHF, VHF, and HF Radios

The military currently employs transceivers that operate in the high-frequency (HF), very high frequency (VHF), and ultra-high frequency (UHF) radio bands. These transceivers perform many functions, including: voice communications, data transmission, and automatic direction finding. A number of different modulation and signal-transmission formats are employed, including: amplitude modulation (AM); frequency modulation (FM); and upper-, lower-, and double-side band (USB, LSB, and DSB, respectively) with suppressed carrier transmission. Generally, these signals can either be sent in the clear or secured through the use of an encrypting device such as the KY-28. The VHF radio bands are primarily used for voice communications/navigation with commercial airfields.

There are many different equipment sets that perform these functions, and more will probably be developed between now and the late 1990s. Also, the particular radios required will, to a certain degree, depend upon specific missions. One of the radios could be similar to the ARC-182, which operates in both the UHF and VHF bands. One advantage to this radio is that it is about the same size as radios that typically operate in the UHF band only.

## 6.7 Level-1 Approach to Radio Navigation and Communications

Two major approaches were considered in this investigation. The first, Level 1, is discussed in this section. Level 1 is a modification to the black-box approach that is employed in most current aircraft. Simply stated, it consists of separate systems that are generally developed by independent program offices, the outputs of which are integrated onboard the subject aircraft. A number of alternatives are discussed. The primary difference between these alternatives is the division of processing between the fault-tolerant multiprocessor (FTMP), regional processors, and processors embedded in the systems. Further studies are necessary to determine which of these alternatives is best. Navigation and communications are discussed separately in Sections 6.7.1 and 6.7.2.

### 6.7.1 Radio-Navigation Configuration Alternatives

Within the Level-1 approach, three basic alternative configurations were developed for radio navigation. The systems considered were GPS, JTIDS, TACAN, and an onboard radio system capable of providing estimates of vehicle velocity along and cross track. (This could be either a Doppler radar or a correlation velocity sensor.)

#### 6.7.1.1 Alternative 1

The first alternative is depicted in Figure 6-6. It is composed of a GPS receiver, two JTIDS terminals, and a velocity sensor. There are two JTIDS terminals in the avionics suite, primarily to provide redundant communications. They also can be used to provide redundant JTIDS relative navigation and TACAN.

The functions performed by the units labeled JTIDS are those performed by the transceiver and signal processor described in Section 6.3.2 and Appendix 6-B.

The unit labeled GPS performs the functions of the GPS receiver, signal processor (process controller), and the data processor, except for the functions referred to as navigation filter and navigation in Appendix 6-A.

The velocity sensor supplies measurements of vehicle velocity along and cross track.

The following functions are performed at a regional processing level.

##### (1) TACAN

This function is responsible for processing TACAN beacon signals in order to provide bearing and distance information. This is described in greater detail in Appendix 6-B.

##### (2) TACAN Fault Detection and Identification (FDI)

This routine takes the range (R) and bearing ( $\theta$ ) estimates independently generated from the signals from the two JTIDS units, and determines which, if either, of these two sets of estimates should drive the displays. It does this by comparing the two estimates with each other and with predetermined estimates of range and bearing. These predetermined estimates are generated from onboard estimates of vehicle position and heading and the known position of the TACAN transmitter (in the case of air-to-air or ship-to-air TACAN, the position of the TACAN transmitter is known because that information is communicated over the JTIDS data link). This routine also makes use of the fault indicators (FIs) of the JTIDS units and the status indicators (SIs) provided by the JTIDS time-of-arrival (TOA) and communications FDI routines. In addition, this routine determines which of the JTIDS units should transmit TACAN. As



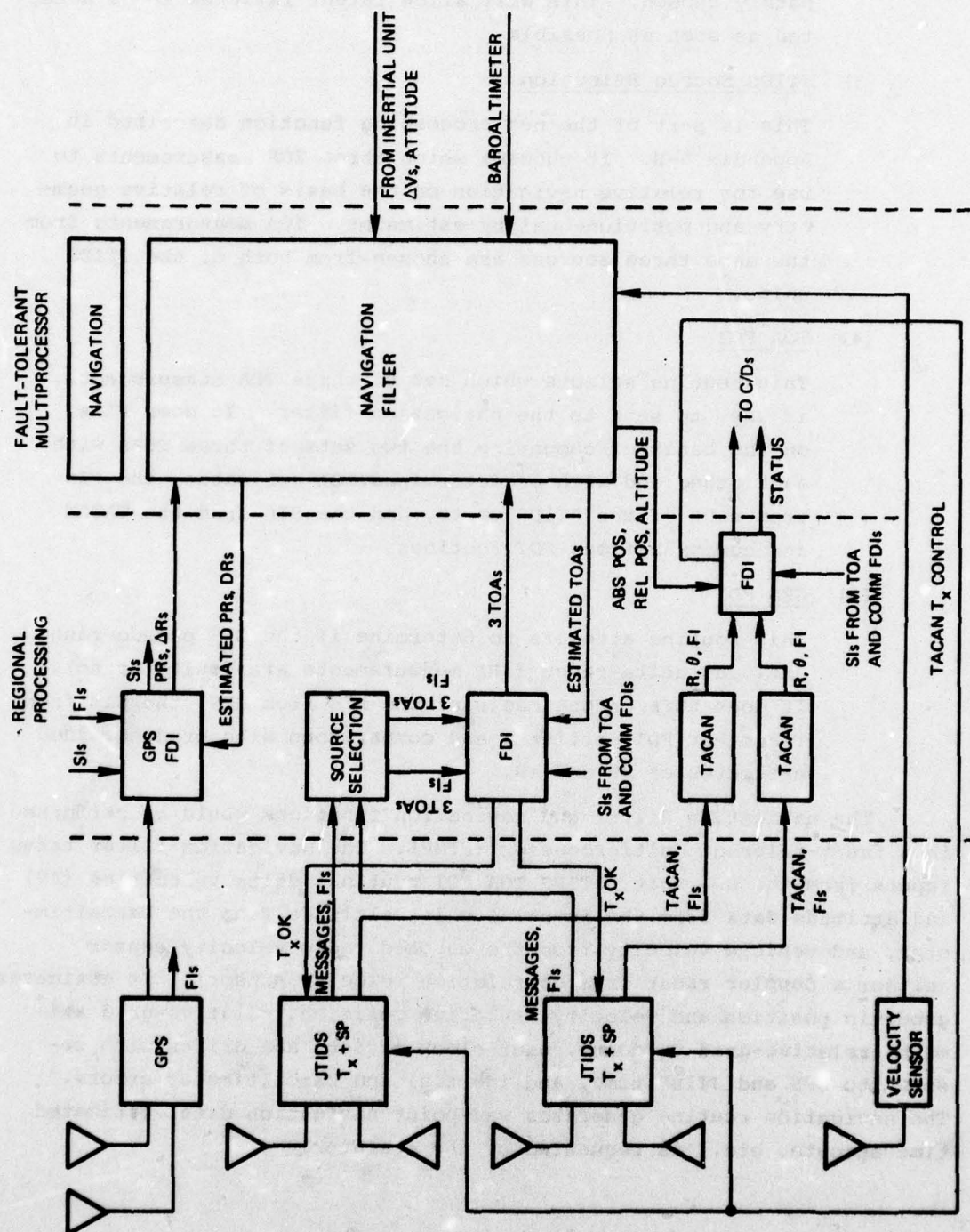


Figure 6-6. Radio-navigation configuration alternative 1.

long as both transmitters are working, they will be alternately chosen. This will allow latent failures to be detected as soon as possible.

(3) JTIDS Source Selection

This is part of the net processing function described in Appendix 6-B. It chooses which three TOA measurements to use for relative navigation on the basis of relative geometry and position-quality estimates. TOA measurements from the same three sources are chosen from both of the JTIDS units.

(4) TOA FDI

This routine selects which set of three TOA measurements, if any, to send to the navigation filter. It does this on the basis of comparing the two sets of three TOAs with each other and with predetermined TOA estimates, the FIS from each of the JTIDS units, and the SIs from the TACAN and communications FDI routines.

(5) GPS FDI

This routine attempts to determine if the GPS pseudo-range (PR) and delta-range ( $\Delta R$ ) measurements are faulty or not. It does this on the basis of the FIS from GPS, the SIs from the other FDI routines, and comparisons with predetermined estimates of PR and  $\Delta R$ .

The navigation filter and navigation functions would be performed in a fault-tolerant multiprocessor (FTMP). The navigation filter takes inputs from the GPS unit, JTIDS TOA FDI routine, delta velocities ( $\Delta V$ ) and attitude data from the inertial unit, altitude from the baroaltimeter, and vehicle velocity from the onboard radio velocity sensor (either a Doppler radar or a correlation velocity sensor). It estimates geodetic position and velocity, relative position, relative-grid azimuth, relative-grid velocity, user-clock offsets and drifts with respect to GPS and JTIDS time, and inertial and baroaltimeter errors. The navigation routine generates way-point navigation data, estimated time enroute, etc., as requested by the operator.



#### 6.7.1.2 Alternative 2

Alternative 2 is depicted in Figure 6-7. The major difference between Alternatives 1 and 2 is that in Alternative 2 the navigation filter and navigation routines are done at a regional processing level, instead of in the FTMP.

#### 6.7.1.3 Alternative 3

Alternative 3 is depicted in Figure 6-8. The major difference between Alternatives 3 and 1 is that the functions performed at a regional level in Alternative 1 are performed in the FTMP in Alternative 3.

#### 6.7.2 Communications Configuration

For presentation purposes, the communications aspect of the Level-1 approach is divided into two areas: tactical-data communications, and voice communications and IFF. The tactical-data communications are discussed in Section 6.7.2.1. Voice communications and IFF are discussed in Section 6.7.2.2.

##### 6.7.2.1 Tactical-Data Communications

The first alternative for the tactical-data communications configuration is shown in Figure 6-9. The recent sea-based C<sup>3</sup> study indicated that JTIDS alone may not be sufficient to handle the tactical-data requirements for V/STOL. (6-9) Thus, a second tactical-data communications system may be required. For this effort, it was assumed that this system could either be a modified Link-4 system, or something similar to JTIDS (depending upon the JTIDS test results, it could be JTIDS operating in a separate net). In either case, the two systems would probably have to be synchronized, and would transmit complementary data. Two units of each system would be required to provide fault-detection capability. All of the processing with the exception of direct C/D processing would be performed at the regional processing level. See Section 6.3.2 and Appendix 6-B for a discussion of the JTIDS data processing. In addition, this regional processor performs a communications FDI function. An FDI routine takes the messages from the two JTIDS units (which are discussed in the previous section) and compares them. If they disagree, the routine attempts to identify which one is incorrect. It does this on the basis of FIs from the units, and SIs

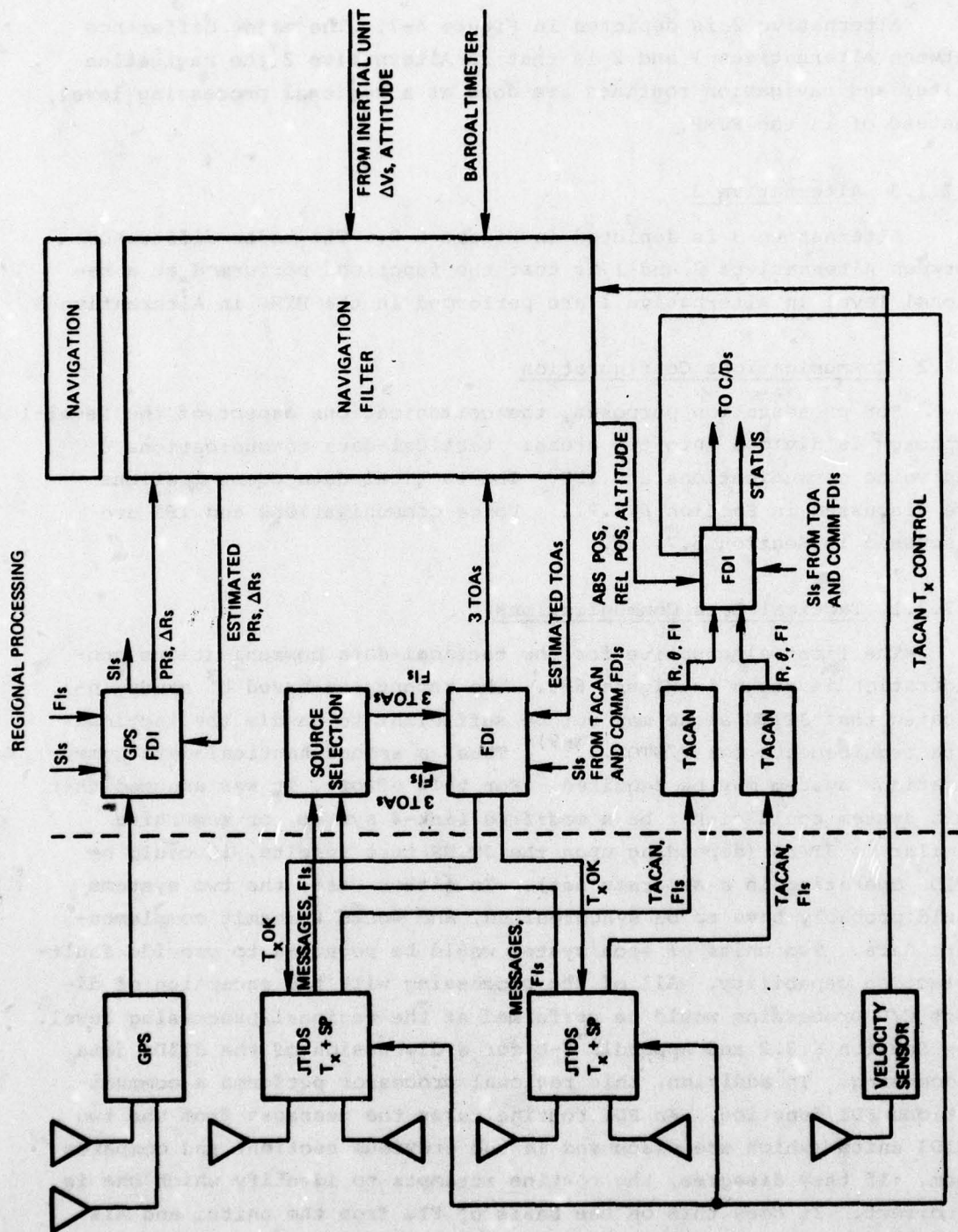


Figure 6-7. Radio-navigation configuration alternative 2.





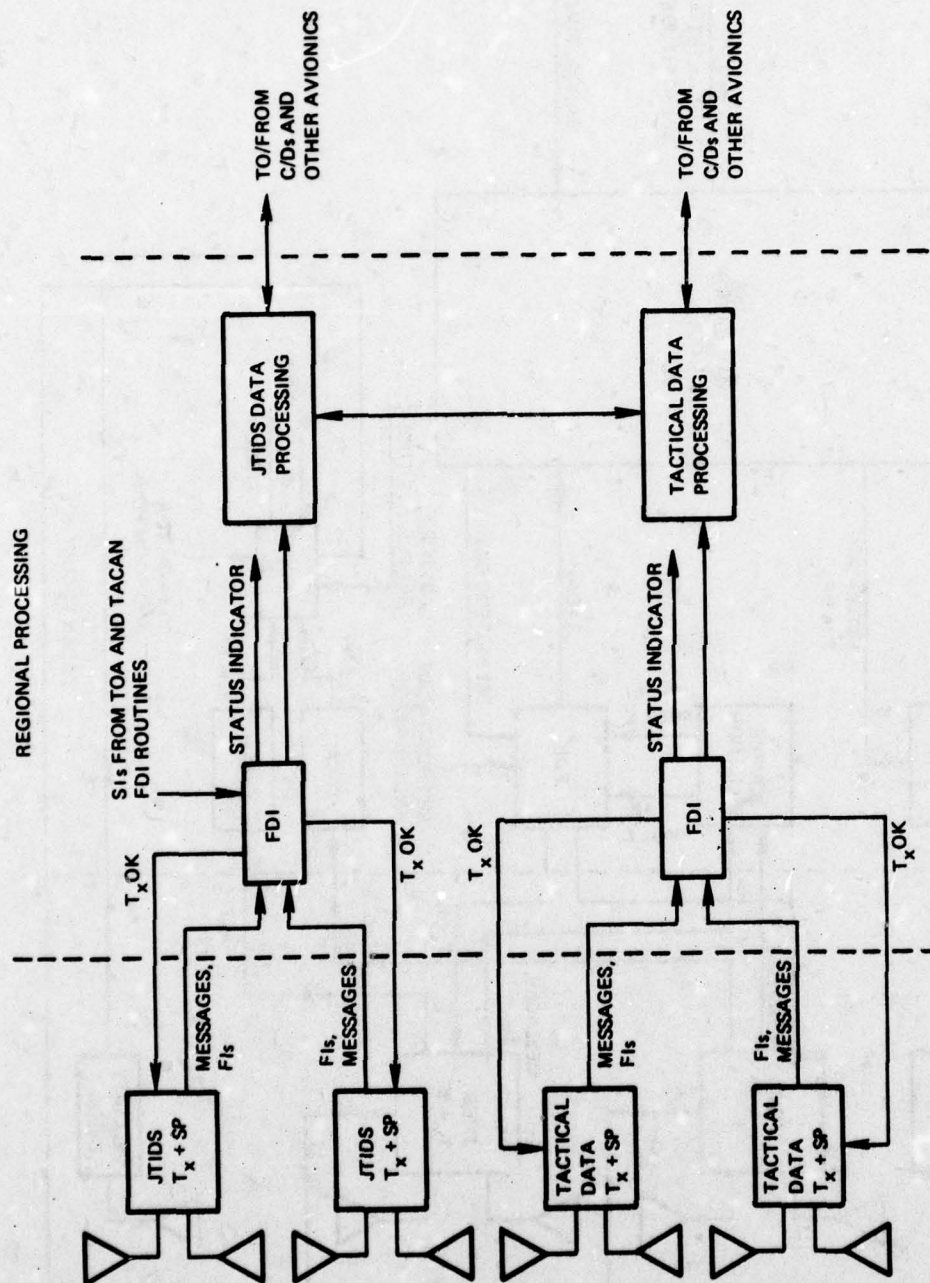


Figure 6-9. Tactical-data communications alternative 1.



from the TOA and TACAN FDI routines. It could also request that the data be sent over the other data system for an additional comparison if so desired. The communications FDI routine also controls the transmitters of the JTIDS units. The FDI and data processing for the other tactical-data system would be similar to that of JTIDS.

The second alternative is shown in Figure 6-10. In this alternative, the processing is done in the FTMP instead of in regional processors.

#### 6.7.2.2 Voice Communications and IFF

The UHF, VHF, and HF radio and IFF configurations are depicted in Figure 6-11. The FDI routines look at the FIs of each of the radios, and where appropriate compare the receiver outputs. The FDI routines control which of the dual-redundant radios is transmitting, and as long as both units are working, they alternate transmissions between the two to help detect and identify latent failures.

#### 6.8 Level-2 Approach

This section discusses a highly integrated approach to radio navigation and communications. In general, integration is preferred over the black-box approach. If done properly, integration can reduce overall weight, size, life-cycle costs, and power and cooling requirements, while at the same time increasing operational effectiveness and the probability that flight and/or mission-critical functions will be available when required. This section deals primarily with the Tactical Information Exchange System. It is about to enter its first phase of development. The current TIES concept and its applicability to an integrated fault-tolerant avionics system for future tactical aircraft are discussed in the following sections. Navigation and communications are again discussed separately (see Sections 6.8.2 and 6.8.3).

##### 6.8.1 Tactical Information Exchange System (TIES)

The TIES program is an attempt to develop a functionally integrated Communications, Navigation, and Identification (CNI) system. (6-10, 6-11, 6-12) It is intended to be employed in the 1990s time-frame with particular concern for the V/STOL application. It is just about to enter the first

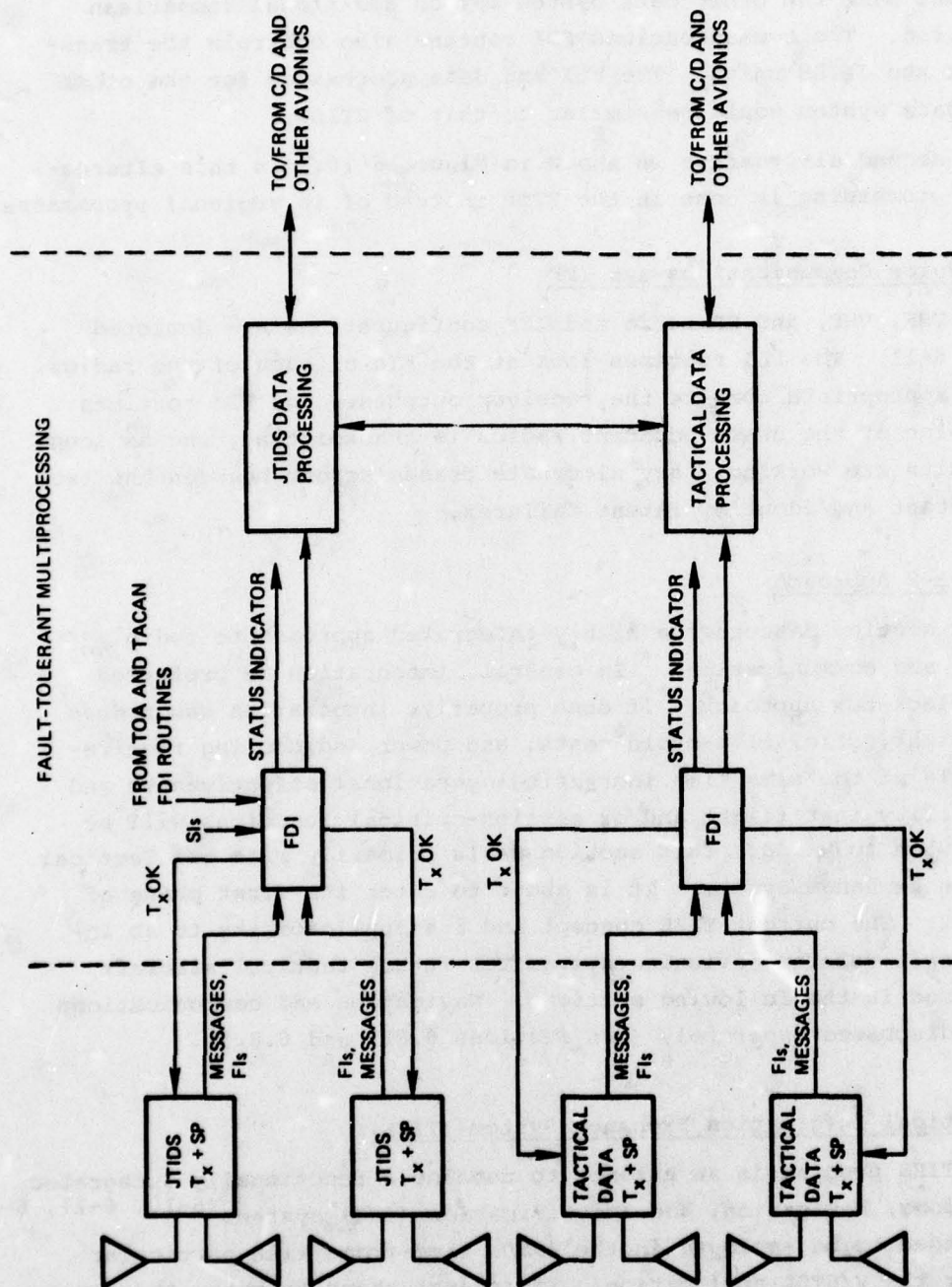


Figure 6-10. Tactical-data communications alternative 2.



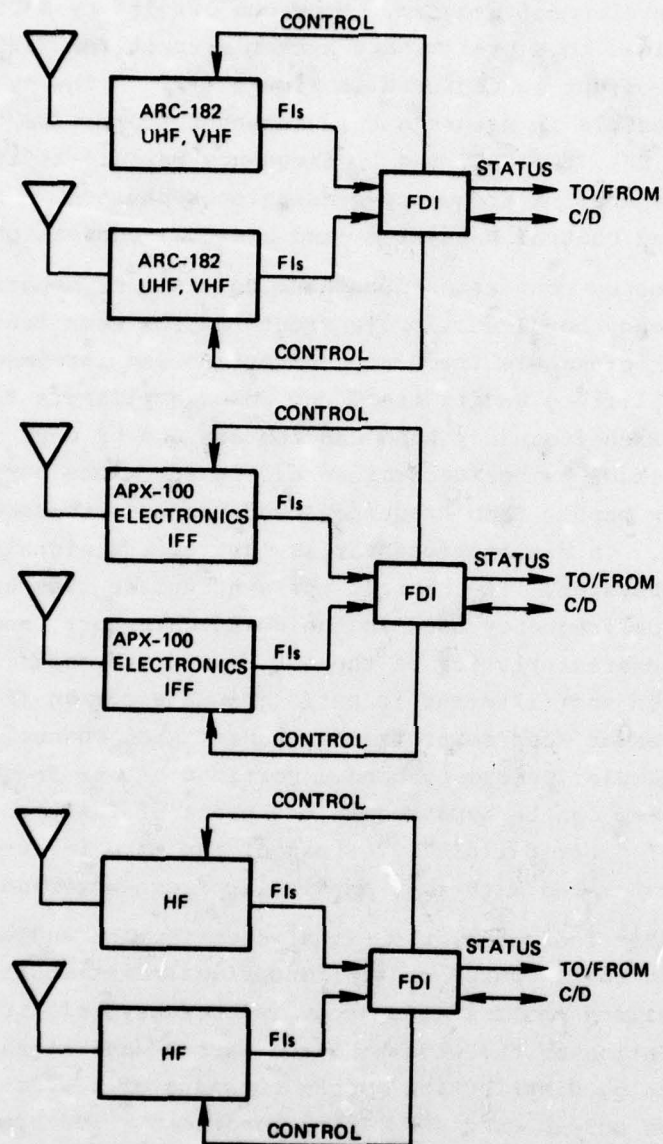


Figure 6-11. Voice communications and IFF configurations.

phase of its development program. Previous efforts by NADC and contractors have lead to a preliminary system concept for TIES. The preliminary TIES concept is depicted in Figure 6-12.\* The system is modular and flexible in nature and is intended to receive/transmit signals in the HF, VHF, UHF, and  $L_x$  frequency bands. It is composed of three subsystems: a frequency-conversion subsystem, a signal-distribution and control subsystem, and a signal-conversion subsystem.

The frequency-conversion subsystem consists of separate antennas for each frequency band/function, a front end for each frequency band, multipurpose programmable frequency converters and intermediate frequency (IF) amplifiers and filters, and power amplifiers for each frequency band. Each frequency band can contain one or more receiver channels with the exact number determined by the functions performed in that particular band. Each frequency band has a microprocessor associated with it. This microprocessor is part of the signal-distribution and control subsystem. It controls the synthesizer that generates the local oscillator frequency used in the down convertors, and the gain and bandwidth characteristics of the IF amplifiers and filters. All IF amplification and filtering is performed at a common IF of 70 MHz. This microprocessor also controls the transmission channels associated with that particular frequency band. Portions of the frequency conversion subsystem can be bypassed in the event of failure. This provides a form of "soft failure", instead of the hard failures that would normally be associated with that particular frequency band.

As its name indicates, the signal-distribution and control subsystem provides total system control and signal distribution. In addition to controlling the frequency-conversion subsystem, it also controls the operation of the wide-band and narrow-band signal-conversion units. The analog distribution system consists of a wide-band frequency division multiplexed (FDM) bus (bandwidth > 500 MHz) and associated coupling devices. The signal-distribution subsystem also handles digital data at the rates of 1 and 10 MHz. The different rates are handled on separate buses to reduce costs. This subsystem is also responsible for distributing the time and frequency standards for the system. Both the receive and transmit portions of the wide-band FDM bus are dual-redundant, and can be tied together for subsystem testing.

---

\* This figure and some of the following information was obtained from NADC personnel working on the TIES program.



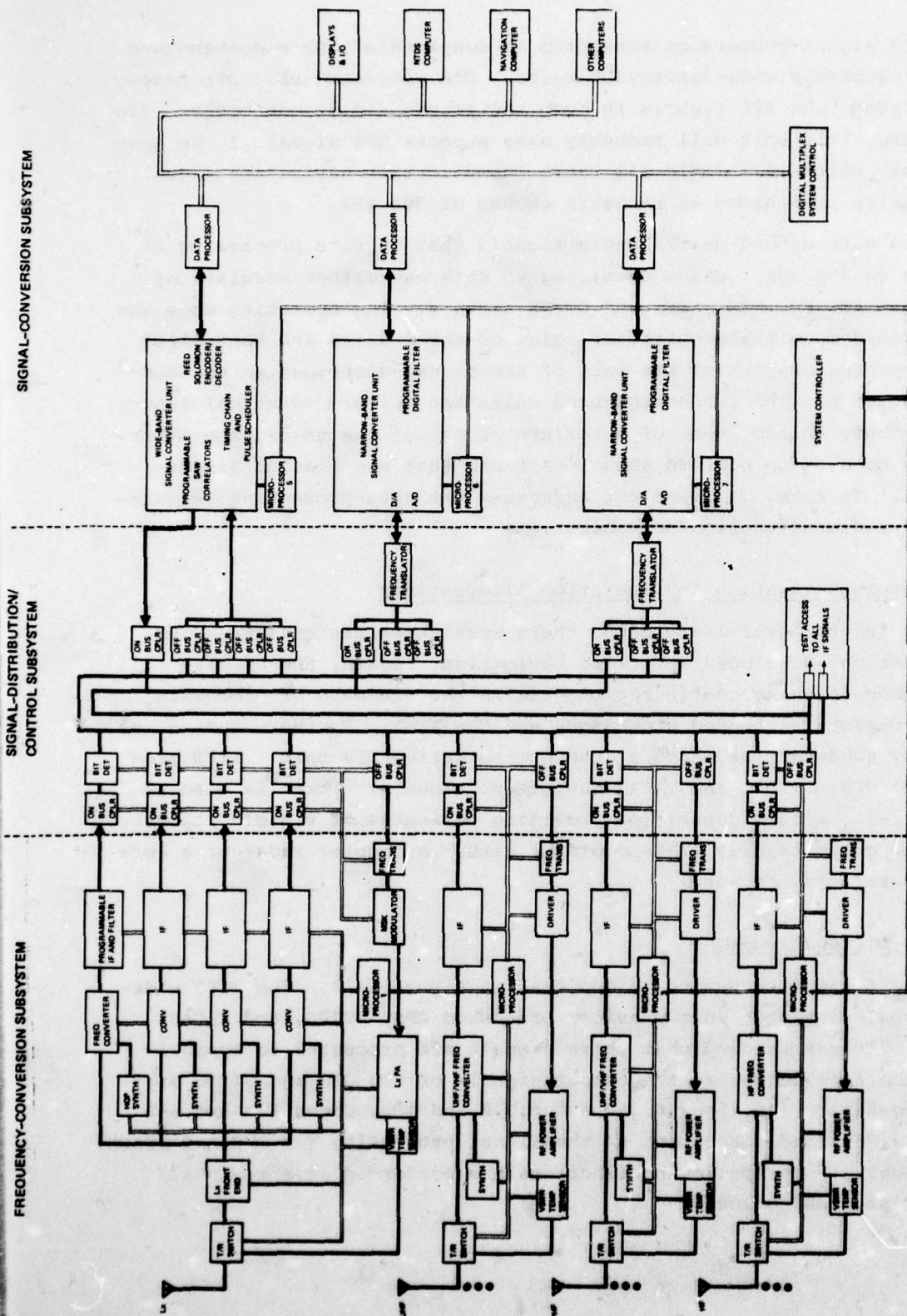


Figure 6-12. TIES concept. \*

\* This figure was obtained from NADC personnel working on the TIES program.

The signal-conversion subsystem is composed of one wide-band and two narrow-band signal-conversion units. The wide-band unit processes JTIDS, TACAN, and IFF signals in both the encode and decode modes. (In the future, this unit will probably also process GPS signals.) In general, this unit will handle all radio communication/navigation signals that require processing at rates in excess of 300 kHz.

The narrow-band units handle signals that require processing at rates below 300 kHz. Units developed to date can either modulate or demodulate AM, FM, FSK, SSB, and DCPSK signals. The operating mode and modulation/demodulation characteristics of these units are controlled by microprocessors, which are part of the signal-distribution and control subsystem. The two narrow-band units can perform identical functions. Thus, in the event of a failure of one of the units, the other unit can be used to perform those functions that are mission/flight-critical. To date, TIES has not addressed the data-processing requirements of radio navigation/communications.

#### 6.8.2 Radio-Navigation Configuration Alternatives

As in the Level-1 approach, there were three basic alternative configurations developed for radio navigation. Again, the primary differences in these configurations lie in the division of processing between regional/embedded processors and the FTMP. Further studies are necessary to determine which of these alternatives is best. TIES provides the GPS, JTIDS, and TACAN navigation signals. There is also an onboard radio system capable of providing estimates of vehicle velocity along and cross track. (This could be either a Doppler radar or a correlation velocity sensor.)

##### 6.8.2.1 Alternative 1

The first alternative is depicted in Figure 6-13. The TIES wide-band signal-converter unit provides processed GPS, JTIDS, and TACAN signals. It was assumed that these signals are processed to approximately the same degree as the output signals of the GPS and JTIDS signal processors. (Sections 6.2.2 and 6.3.2 and Appendices 6-A and 6-B provide a detailed discussion of the signal processing for GPS and JTIDS, respectively.) The following functions are performed at a regional/embedded processing level.



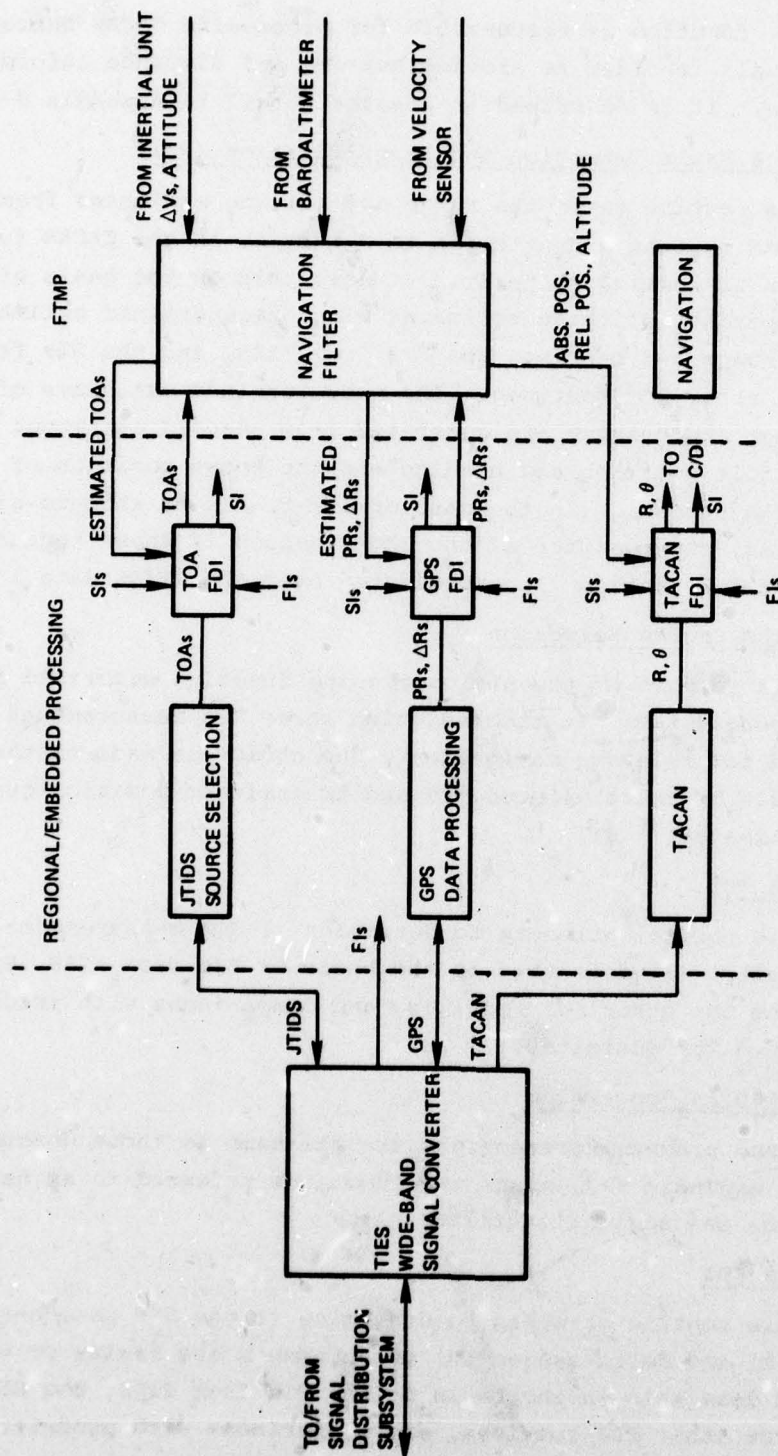


Figure 6-13. Integrated TIES approach to radio navigation alternative 1.

(1) TACAN

This function is responsible for processing TACAN beacon signals in order to provide bearing and distance information. It is described in greater detail in Appendix 6-B.

(2) TACAN Fault Detection and Identification (FDI)

This routine takes the range and bearing estimates from the TACAN routine and attempts to determine if the TACAN function is working properly. It does this on the basis of a comparison of these estimates with predetermined estimates of range and bearing, the FIs from TIES, and the SIs from the other FDI routines. The predetermined estimates of range and bearing are generated from onboard estimates of vehicle position and heading and the known position of the TACAN beacon. (In the case of air-to-air or ship-to-air TACAN, the position of the TACAN beacon is known because that information is communicated over the JTIDS data link.)

(3) JTIDS Source Selection

This is part of the net processing function described in Appendix 6-B. It chooses which three TOA measurements to use for relative navigation. The choice is made on the basis of relative geometry and transmitter position-quality estimates.

(4) TOA FDI

This routine attempts to determine if TOA measurements are faulty. It does this on the basis of FIs from TIES, SIs from the other FDI routines, and comparisons with predetermined TOA estimates.

(5) GPS Data Processing

These processing functions are the same as those described in Appendix 6-A, minus the functions referred to as navigation and navigation filter.

(6) GPS FDI

This routine attempts to determine if the GPS pseudo-range (PR) and delta-range ( $\Delta R$ ) measurements are faulty or not. It does this on the basis of the FIs from TIES, the SIs from other FDI routines, and comparisons with predetermined estimates of PR and  $\Delta R$ .



The navigation filter and navigation functions are performed in the FTMP. The navigation filter takes inputs from the GPS unit, JTIDS TOA FDI routine, delta velocities ( $\Delta V$ ) and attitude data from the inertial unit, altitude from the baroaltimeter, and vehicle velocity from the onboard radio velocity sensor (either a Doppler radar or a correlation velocity sensor). It estimates geodetic position and velocity, relative position, relative grid azimuth, relative grid velocity, user-clock offset and drifts with respect to GPS and JTIDS time, and inertial and baroaltimeter errors. The navigation routine generates way-point navigation data, estimated time enroute, etc., as requested by the operator.

#### 6.8.2.2 Alternative 2

Alternative 2 is depicted in Figure 6-14. The major difference between alternatives 1 and 2 is that in alternative 2 the navigation filter and navigation routines are done at a regional/embedded processing level, instead of in the FTMP.

#### 6.8.2.3 Alternative 3

Alternative 3 is depicted in Figure 6-15. The primary difference between alternatives 3 and 1 is that in alternative 3 all the processing is performed on the FTMP.

#### 6.8.3 Integrated Approach to Communications

The communications configuration for alternative 1 is depicted in Figure 6-16. The signals processed are JTIDS, IFF, and the HF, VHF, and UHF voice and teletype signals. (As discussed previously, there may have to be an additional tactical-data system to complement JTIDS; it is not shown in this figure because of the uncertainty of its nature.) The JTIDS output of the wide-band signal converter is assumed to be processed to approximately the same degree as the output of the JTIDS signal processor discussed in Section 6.3.2 and Appendix 6-B. All other signals are assumed to be processed to the same degree as the signals provided by the appropriate black boxes described in Sections 6.5 and 6.6.

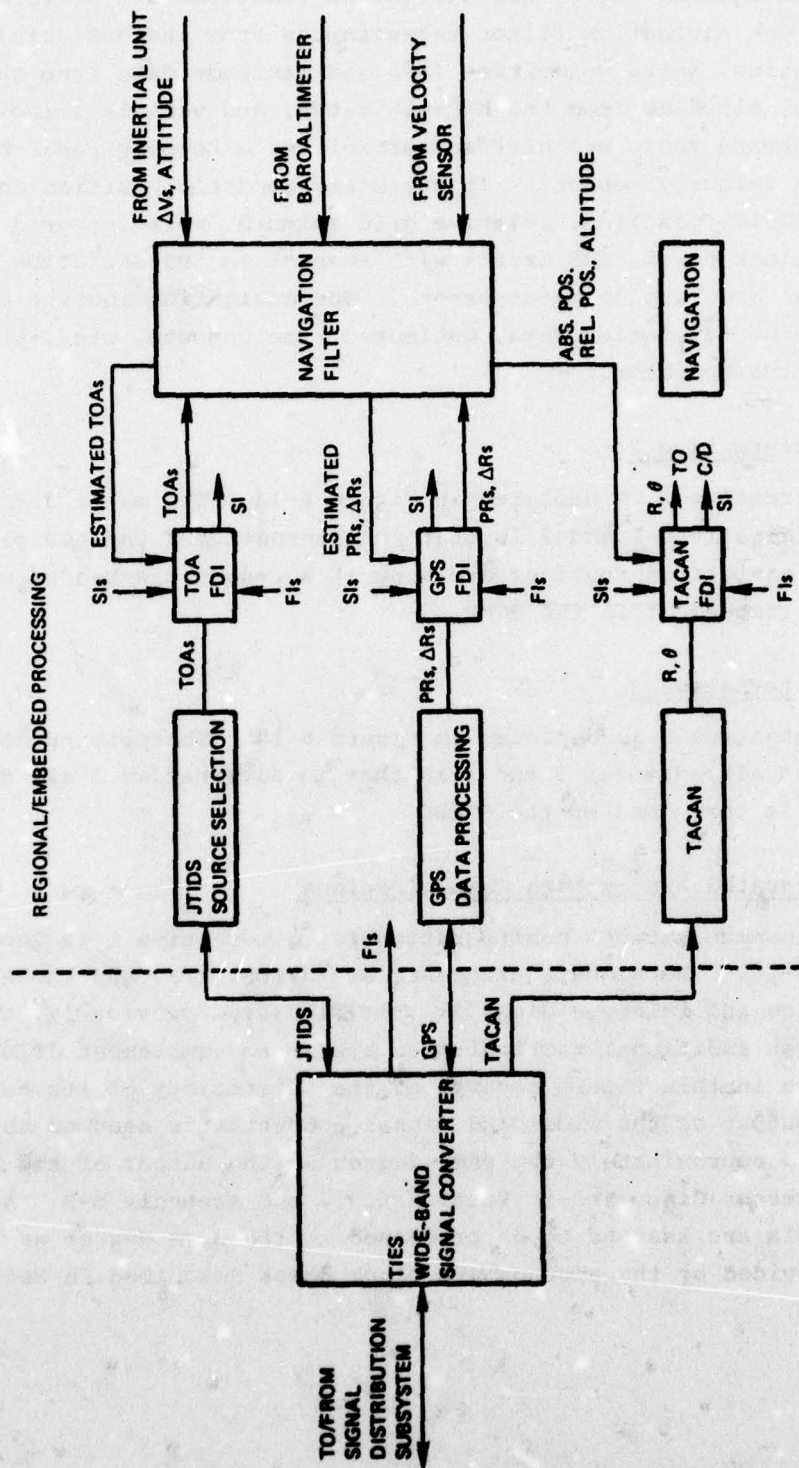


Figure 6-14. Integrated TIES approach to radio navigation alternative 2.



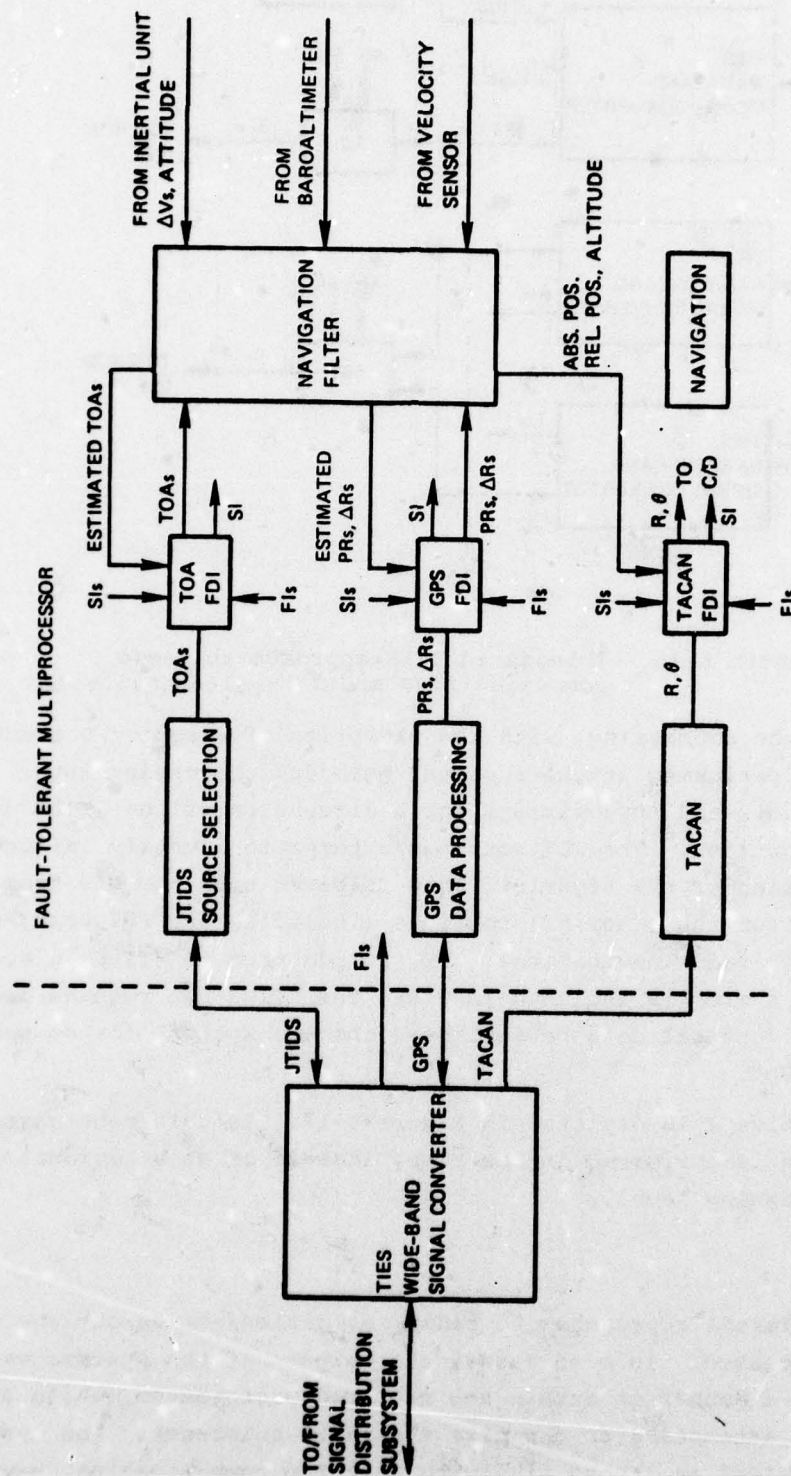


Figure 6-15. Integrated TIES approach to radio navigation alternative 3.

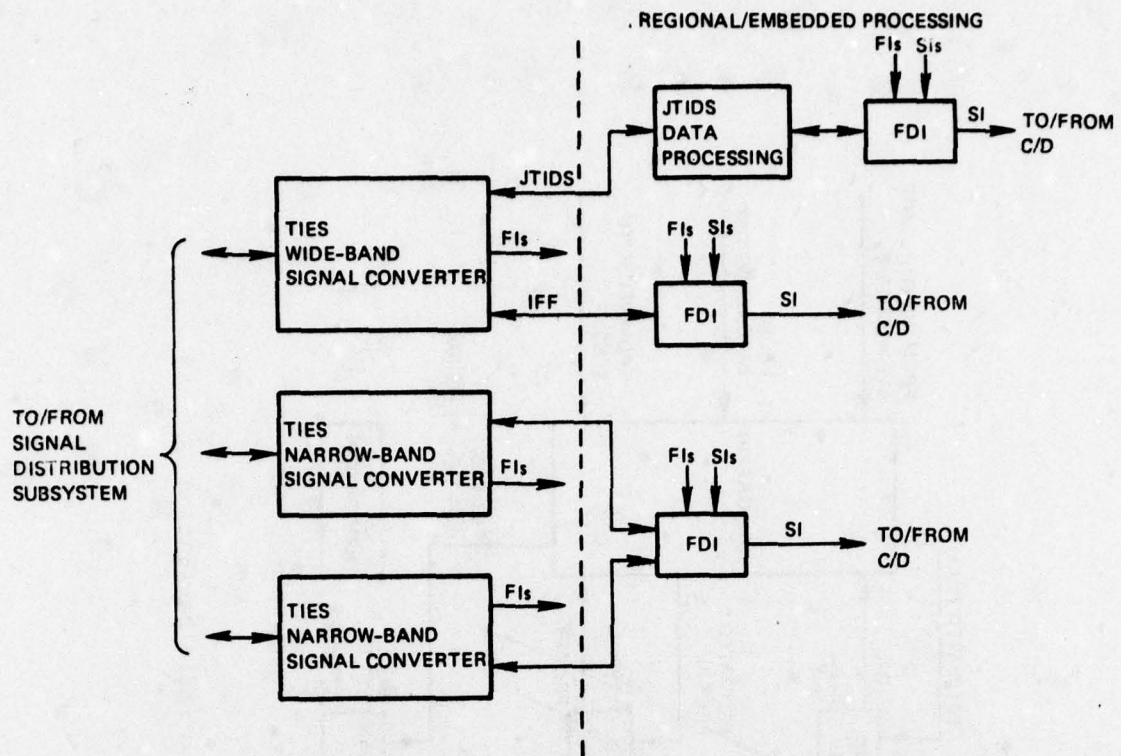


Figure 6-16. Integrated TIES approach to radio communications and IFF—alternative 1.

All of the processing, with the exception of direct C/D processing, would be performed at the regional/embedded processing level. See Section 6.3.2 and Appendix 6-B for a discussion of the JTIDS data-processing functions. The FDI routines attempt to identify failures in the processing of the signals. This is based upon the FIs from TIES and SIs from the other FDI routines (including the FDI routines associated with radio navigation). If, in addition to JTIDS, a second tactical-data system is included in TIES, the JTIDS FDI routine could request that identical data be sent over the two systems for comparison as an aid in FDI.

Alternative 2 is depicted in Figure 6-17. In this configuration the processing is performed in the FTMP, instead of at a regional/embedded processing level.

#### 6.9 Summary

Two different approaches to radio-navigation/communications systems were discussed. In both cases, the outputs of the systems were integrated in a manner to attain the required performance, while at the same time attempting to maximize the fault tolerance. The system functions provided are those likely to meet the communication, navigation, and identification requirements of a tactical aircraft in the



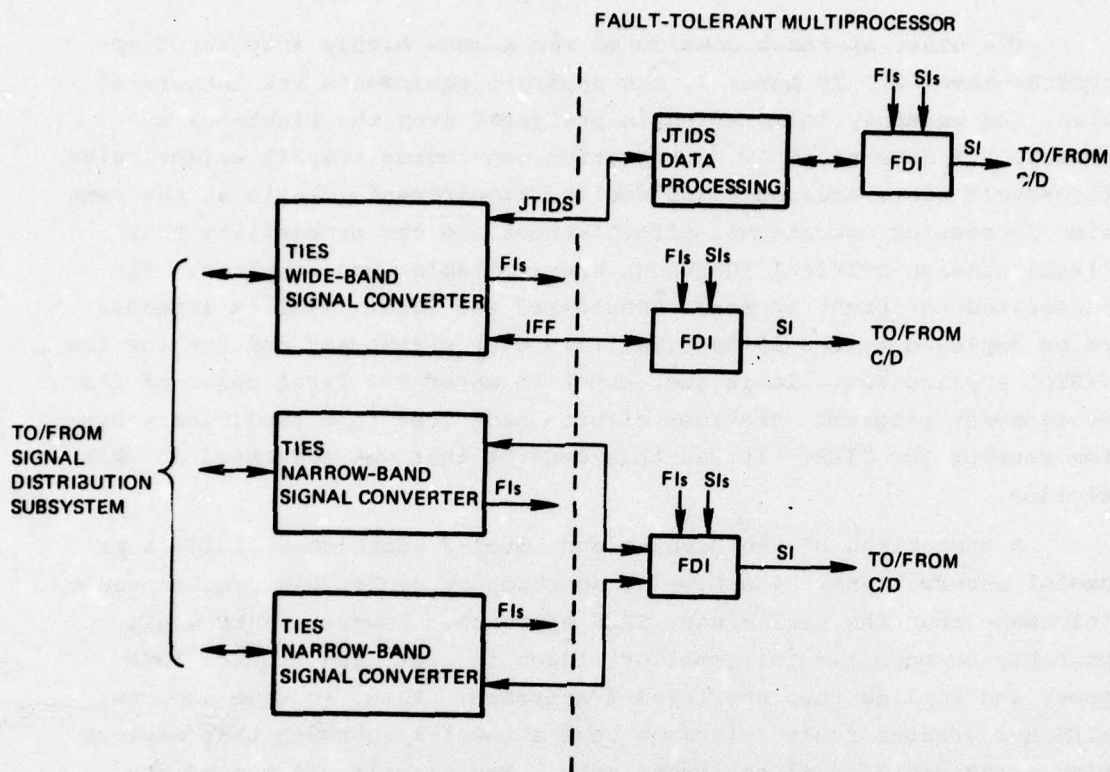


Figure 6-17. Integrated TIES approach to radio communications and IFF—alternative 2.

late 1990s. The communications requirement can be divided into two areas: tactical-data transfer (generally secure), and voice communications with both members of the military (both secure and clear) and with the civilian community. The communications requirements are likely to be met using JTIDS (a second tactical-data system possibly similar to JTIDS may be required) and UHF, VHF, and HF radios, and (where appropriate) the associated encrypting units. The navigation requirements can also be divided into two areas: relative/area, and geodetic navigation. The relative/area navigation is performed both with respect to military and civilian reference points. The systems considered for navigation were, JTIDS, TACAN, GPS, VOR, and an on-board radio velocity sensor (such as either a Doppler radar or a correlation velocity sensor). It was assumed that the identification function would be performed by a typical IFF system.

The first approach addressed was a modification to the one employed in most current aircraft. Referred to as the Level-1 approach, it consists of separate systems that are generally developed by independent program offices whose outputs are integrated onboard the subject aircraft.

The other approach considered was a more highly integrated approach—Level 2. In Level 2, the separate equipments are integrated also. In general, integration is preferred over the black-box approach. If done properly, integration can reduce overall weight, size, life-cycle cost, and power and cooling requirements, while at the same time increasing operational effectiveness and the probability that flight/mission-critical functions are available when required. The integrated-equipment approach considered was TIES. TIES is intended to be employed in the 1990s time frame with particular concern for the V/STOL application. It is just about to enter the first phase of its development program. Previous efforts have lead to a preliminary system concept for TIES. It was this concept that was addressed in this section.

A comparison of the Level-1 and Level-2 approaches yields some useful observations. The Level-1 approach presented has greater fault tolerance than the preliminary TIES approach. However, TIES would probably be much lighter, smaller, lower in cost, and require less power and cooling than the Level-1 approach. Also, in some aspects, TIES has greater fault tolerance than a Level-1 approach that employs single instead of dual equipment sets. For example, if one of the TIES narrow-band signal-converter units fails, the other can be used either to perform all the functions previously performed by the two units in a degraded performance mode, or to perform only those functions that are flight/mission-critical.

Future TIES design efforts could very well lead to a system that has a fault tolerance that is equivalent to or greater than that of the Level-1 approach presented here. There are some portions of the current TIES concept which should be reviewed when TIES is expanded to include its full complement of systems (e.g., GPS is not part of the preliminary TIES concept). In this review, particular emphasis should be placed upon increasing the overall fault tolerance of the system. Clearly, there will be a tradeoff between fault tolerance and increased size, weight, cost, power, and cooling. Also, some of the functions will not require the same degree of fault tolerance as others.

One concept that should be reviewed places the frequency-conversion subsystems at the antenna sites. It may be advantageous to leave the transmitting portions of these subsystems at the antennas. However, if the frequency converters and IF amplifiers and filters were located



in the equipment racks, they could be used as a bank of receiver channels that could be switched between the various inputs from the different front ends. For example, if a receiver channel being used to demodulate UHF signals failed and that function were considered to be flight/mission-critical, then another receiver channel could be switched to perform that function. This would cause a reduction in performance or, in the worst case, the elimination of a function that is not considered to be flight/mission-critical. The definition of flight/mission-critical will be different for different missions and will change depending upon what stage of the mission the vehicle is in. This concept of modularity would make TIES more flexible and more fault tolerant by making maximum use of the multipurpose receiver channels already developed.

Within both the Level-1 and Level-2 approaches, a number of alternatives were described. Differences between these alternatives are primarily in the area of how the various information-processing tasks will be partitioned between embedded processors, the FTMPs, and possibly the regional processors. The baseline information-processing system described in Section 3.5 contains two FTMPs for damage tolerance. One of these performs all flight-critical functions such as flight control, system management, and network management. Unless a major battle damage event eliminates one of these, the second FTMP represents a significant information-processing resource. It may be possible to perform all the communications and navigation information processing within this second FTMP and within embedded processors, in which case, regional processors will not be required. If, however, it is not practical, regional computers would be incorporated into the system to provide the processing that involves a number of communications elements. This processing must be performed at a fault-tolerant level to permit (at a minimum) fail operational performance. This implies that either the regional processors should be fault-tolerant, possibly a smaller version of the FTMP, or a pool of regional processors would be included in the system to allow flexible allocation of tasks among a number of regional processing units. Knowledge and experience is insufficient at the present time to permit the resolution of this question. Furthermore, the communications area is highly mission-dependent; thus, the configuration may differ significantly from one type of aircraft to another. Additional studies should be performed in this important area.

#### LIST OF REFERENCES

- 6-1 NAVSTAR Global Positioning System (GPS), Concept of Operations, Approved by B. Parkinson, Colonel, US Air Force, Deputy for Space Navigation Systems.
- 6-2 System Specification for the NAVSTAR Global Positioning System, Phase II, SS-GPS-200, 1 September 1977.
- 6-3 System Segment Specification for the User System Segment NAVSTAR Global Positioning System, Phase II, SS-US-200, 1 September 1977.
- 6-4 Stonestreet, William M., A Functional Description of the NAVSTAR GPS Receiver Model X, CSDL Report R-981, Volume I, 26 April 1976, revised February 1977.
- 6-5 Workman, Billy Joe, An Overview of the Joint Tactical Information Distribution System JTIDS, Mitre Report MTR-3228, April 1976.
- 6-6 System Segment Specification for JTIDS Class 2 Terminal, DBC78S3000, 21 August 1978.
- 6-7 Stonestreet, William M., et al, GPS/JTIDS/INS Integration Study Final Report, CSDL Report R-1151, May 1978.
- 6-8 Kayton, M. and Fried, W., Avionics Navigation Systems, John Wiley and Sons, Inc., 1969.
- 6-9 Sea-Based Command, Control, and Communications (C<sup>3</sup>) Study, Op-9.
- 6-10 Palatucci, G.J. and Ressler, E.R., TIES Tactical Information Exchange System, AIAA Paper 77-1498.
- 6-11 TIES Software Orientation and Requirements, NADC.
- 6-12 Statement of Work TIES Modular Packaging Study.



## SECTION 7

### INTERNAL DATA COMMUNICATIONS

#### 7.1 Introduction and Definition

The word "network" can be used in the avionic system context in at least two different senses. The first is the generic sense, in which reference is made to any form of data linkage among sensing and effecting elements, data terminals, and processors.

The second sense of the word network, as is intended here, refers to a specific form of data linkages. In this sense, a network is a communications structure composed of bus segments, or links, which are terminated and interconnected at nodes. The distinction between such a network and more conventional data-bus and dedicated-path communication structures is clarified in Figure 7-1.

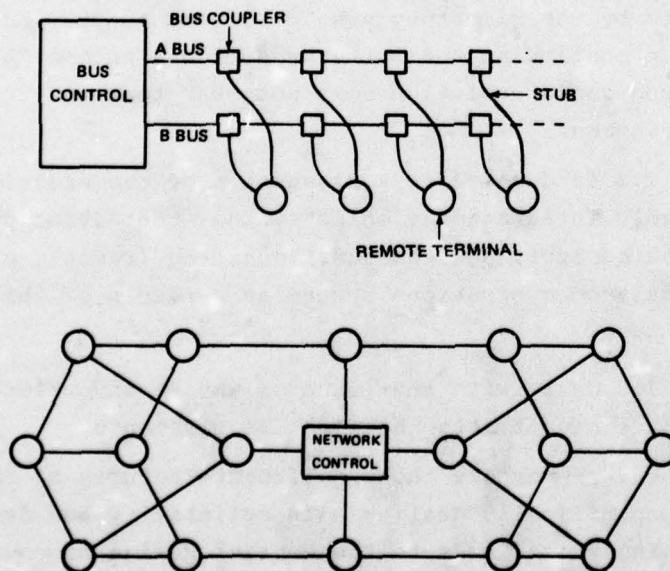


Figure 7-1. Network versus dual-bus structure.

The context in which such a network might be used, and the technical developments and foundation for such a development effort, deserve comment. Clearly, for any complex system to be viable, it is necessary that its architecture be able to incorporate the predecessor devices and technologies, as well as the most current or state-of-the-art technologies. Additionally, it is hoped that a reasonable growth path be available for inserting future devices or technologies. This tends to produce nonhomogeneous designs. The network architecture baseline proposed for internal data communication will, therefore, necessarily support some dedicated-path communications and conventional data busing. It is particularly important that many of the advantages derived from the MIL-STD-1553 data-bus technologies be captured and built upon. Similarly, previous technological successes should not excessively fetter technological evolution. Even though the currently evolving family of 1553 applications is based upon engineering design and devices which have not yet reached deployed maturity, critical shortcomings for certain future applications have already become apparent. Additionally, advanced communication concepts increasingly make existing techniques appear more pedestrian, a phenomenon which is often confused with obsolescence. This sequence of events is common to all rapidly advancing technologies, and requires careful management to avoid floundering. The computer industry as a whole is now well attuned to products becoming obsolete by the time they are introduced to the market, and even with production continuing until the product has become "hopelessly obsolete". Such rapid evolution does not mean that the technologies are beyond management.

Section 7.2 is devoted to a discussion of the underlying concepts of a highly integrated avionics system, the advantages offered by such an architecture, and the additional requirements placed on the internal data-communications system as a result of this new avionics architecture.

Section 7.3 deals with the issue of why a conventional MIL-STD-1553 architecture cannot meet these new requirements.

Section 7.4 summarizes the significant features of the proposed architecture, specifically dealing with reliability and design options and the reasoning which leads to the current design proposal.

Section 7.5 details design specifics for an embodiment of the proposed architecture.

Section 7.6 is a summary.



## 7.2 Highly Integrated Avionics System: Impact on Communications

An avionics system can be defined as the aggregation of elements (sensors, processors, and effectors) which mechanize a particular set of flight-related functions (e.g., navigation, flight control, displays, and controls). Associated with each of these functions is a particular subset of the system elements. If these functions are mechanized as autonomous subsystems, then the subsets are made disjoint, and the interconnectivity problem at the system level is reduced to the interconnection of relatively few numbers of relatively large aggregates of functionally related elements.

In a highly integrated avionics system, on the other hand, the subsets of system elements associated with the various avionics system functions need no longer be disjoint. Indeed, in the terminology of set theory, integration represents an effort to ensure that the total set is as small as possible by allowing the various subsets to intersect or share elements whenever possible. Thus, the problem becomes one of interconnecting relatively numerous small aggregates of system elements.

The interconnectivity problem at the system level for a highly integrated avionics system is thus seen to be fundamentally different from the interconnectivity problem in systems consisting largely of autonomous subsystems. It is this difference which creates new requirements to be met by the communication structure. It is also from the reduction in size and component numbers, the total set size, that the advantages flow.

What are these advantages? First, a significant reduction in weight, volume, and power consumption of avionics systems can be achieved through the multifunctional use of system elements. This multifunctional use consists of using a single set of sensors to satisfy a number of different requirements for a particular kind of measurement, of using a pool of shared information-processing resources to satisfy diverse processing requirements, or of using a small number of effectors in combination to effect a wide variety of control modes. It can be seen from these examples that the multifunctional use of system elements implies that these sensors, processors, and effectors be richly interconnected. Second, since the addition of a single component may in effect add redundancy to several functions, making each more reliable, it is possible to purchase increased reliability very economically. Both of these advantages are very potent in terms of satisfying pressing requirements for reduced size and weight, and for substantial increases in reliability.

What new requirements are placed on the communications structure? Foremost among these new requirements are those demanding ultra-reliability, high bandwidth, and support of many data sources/sinks.

Why is ultra-high reliability now required where it was not before? Previous unintegrated systems employed autonomous subsystems. Even where individual subsystems might be flight-critical, care was taken to minimize or eliminate the flight-safety implications of subsystem-to-subsystem communications failures. While some data were exchanged, which allowed subsystems to optimize their performance, degraded modes or contingency control within the subsystems provided safe control alternatives, even if inter-subsystem communications were to fail. In short, previous designs were working toward reliability requirements derived from maintenance and maintainability goals, logistics costs, and operational convenience and availability goals.

In contrast, failure of the communications within an integrated system has immediate safety implications. Collapse of the communication structure could lead to the loss of the aircraft.

Why must the integrated avionics communications system handle increased data traffic? Previous communications systems designed to handle inter-subsystem data traffic did not see any of the subsystem internal data traffic. For example, the high-bandwidth traffic between the inertial instruments and the navigation computer is not visible external to that subsystem. In a fully integrated system, each of the inertial instruments is a shared resource, and the data traffic between them and the navigation, autopilot, and fire-control functions must be supported. Some of the data traffic within the new avionics architecture is from one source to one target; some is from many sources to a single target; and some is from one source to many targets. Depending on the exact volume and nature of each of these data exchanges, the new architecture must provide dedicated paths, two-way buses, broadcast buses and a hierarchical aggregate of all of these elements. Additionally, it must provide the necessary redundancy and robustness so that the communications structure can survive the random faults, data-terminal failures, and physical or battle damage that cannot be purged from its environment. It must provide all of this with maximum reliability and minimum complexity and flexibility.

Finally, it is clear that when the integrated avionics system is compared to more conventional designs, the numbers of communicating data terminals have increased greatly. Thus, the communication system is dealing with a multiplexing problem made more complex by an increased



number of data sources and sinks. In effect, the integration of the avionics system, through multifunction use of elements and pooling of resources, allows significant reductions in numbers of sensors, displays, processors, actuators, etc. These reductions come at the expense of higher connectivity and reliability requirements for data communications. The tradeoff is highly favorable for the integrated system because of significant recent advances in electronic technology, which have enormously reduced the cost/capability ratio of the required data-communications facilities.

### 7.3 Limitation of Current Bus Architectures

The existing bus standard is MIL-STD-1553B. This standard, its predecessor "A" version, and the various applications of 1553, represent a fairly well accepted architectural framework. This architecture is the current redundant bus architecture or concept.

This existing concept has shown itself to be reasonably resilient to pressure from its various applications. Some of this pressure has created confusion as to connectors, wave forms, specific meaning of various mode and submode commands, and other growth-pain incompatibilities that have diluted the benefits which might have been expected from 1553. Nevertheless, the hybrid microcircuit and large-scale integration (LSI) implementations have proceeded, and it is on these developments that economic viability and practicality will be based. Clearly, the investments required to rival 1553 microcircuit and component developments preclude the development of an unrelated competitive standard for a similar architecture. The incompatibilities will be worked out in the 1553 applications, and many new applications will be able to live reasonably comfortably within this agreed-upon architecture. However, the MIL-STD-1553 architecture is not infinitely expandable or elastic, and there will arise new technological demands which cannot be met. New solutions will have to be found.

The most significant shortcomings of 1553 within the context of a fully integrated avionics system are its inability to interconnect many data terminals, its vulnerability to physical damage, and an inability to assure that a single terminal will not bring down all attached buses due to erroneous transmissions.

The problem of being able to handle only a limited number of terminals (less than 31) has its roots at two sources. First, the twisted-pair, transmission, line-termination, and terminal-coupling

techniques chosen cannot tolerate many more than 30 terminals. Secondly, the protocol allows address space for no more than 31 remote terminals.

Historically, these limitations were the result of an architectural concept that viewed remote terminals as fairly large aggregations of possibly unrelated sensors or actuators. Since each remote terminal handled many sensors or actuators, the terminal limit did not seem to constrain the system significantly.

To realize fully the advantages of integration, however, it is important that the numbers of individual sensors or actuators handled by a remote terminal be kept small. If this is not done, the failure of a single remote terminal can result in the loss of an excessive portion of the system's resources.

This problem has been partially attacked by the use of hierarchical buses. In its most conventional application, several subsystems might be joined by one 1553 bus (or dual bus), and within each subsystem, a 1553 bus (or dual bus) is used to interconnect subsystem components. This solution parallels conventional architectures of separate autonomous subsystems. However, it is sensitive to failure modes which would make all the sensors or actuators of an entire sub-bus unavailable, due to failure of the terminal connecting that sub-bus to its supervisor bus. It also fails to address the case where it is indeed desirable to organize many data terminals onto one bus. This latter case more truly represents the natural organization of a highly integrated system, where one sensor must be used by several functions, rather than by just one subsystem.

It is possible by appropriate use of repeaters or bus buffers to eliminate the electrical constraint on numbers of terminals which can be interconnected, and still maintain functional compatibility with 1553. It is not possible to eliminate the protocol constraint without some modification to 1553.

The bus's vulnerability to damage is a result of the fact that any damage to any portion of the bus can disable the entire bus, and that the bus is distributed widely, thus having a rather broad cross-sectional area to damage. Since a bus with more than one shorted stub is also likely to be disabled, this cross-sectional area to damage must include the stubs and portions of the remote terminals. This vulnerability provides a mechanism whereby fairly local damage can impact distant equipment. Damage to the wing could disable elevator control,



for example. Any design which is truly flight-critical must include damage- and fault-containment mechanisms that are as effective as the fault containment mechanisms of the airframe structure itself.

The essence of the proposed architecture is to eliminate such bus vulnerability, while still maintaining functional compatibility with multiplex bus operation. This could be done while maintaining functional compatibility with the 1553 standard.

The final serious weakness of 1553 is the relatively ineffective mechanism for preventing faulty terminals from talking out of turn and disabling the bus. The preventive mechanisms which are included are partly effective to the extent that this mode of failure is not likely to be a serious maintenance, operational, or diagnostic problem. However, the uncovered failure modes which could result in a "babbling" terminal are adequate to present a serious safety threat. Examples of such failures have already appeared in the field; one in particular resulted in the loss of an entire dual-bus system due to a single fault. It is probably in this particular aspect of the 1553 design that the difference between designing to maintenance and operational goals and designing to flight-critical standards becomes most evident. The basic reliability of the dual 1553 bus design is such that operational and availability impacts on an aircraft due to data-interconnect malfunction should be minimal. The 1553 bus is very sound, easily maintained, and unlikely to cause aborted missions or other operational difficulties. It represents a significant and dramatic improvement over previous practice. However, when a communications failure is magnified from an operational aggravation (such as an aborted mission) to a loss of aircraft, the reliability constraints are increased significantly. Thus, while the cost of two mission-aborts per year per fleet of aircraft is almost invisible in the maintenance and operational costs associated with the fleet, the loss of two aircraft per year is highly visible, particularly if these losses are compounded with loss of life.

The sources of this vulnerability are many. Some of the dual-bus implementations are particularly vulnerable due to designed-in single-point failures. The primary defense, the watch-dog timer on bus activity by a terminal, is ineffective against address decoder failures in the terminal, which cause it to respond to either the wrong address or to all commands. The interaction of a faulty terminal with broadcast modes, or the interactions between dual buses, present fairly simple mechanisms for disabling one or all buses of a

redundant 1553 bus system. All of these mechanisms have likelihoods or probabilities associated with them, which are insignificant if the only costs associated with them were maintenance actions and operational costs, but which are much too large if flight safety is involved.

These weaknesses can be overcome without breaking with functional compatibility with 1553. The same mechanisms used to overcome the bus vulnerability to damage are also effective in overcoming the babbling terminal problem. A proposed solution is outlined in Section 7.4.

Additional weaknesses of MIL-STD-1553, such as inadequate encoding of the data and commands for error recovery and detection, are not serious enough that they could not be designed around or coped with.

#### 7.4 Basic Network Architecture

The proposed architecture baseline for the communication network is a natural evolutionary step beyond 1553 practices. The constituent parts are bus segments (or links) and nodes which terminate and interconnect these links. A virtual bus can be created by activating circuitry within nodes, which effectively connects appropriate bus segments, one to another. This circuitry is analogous to relay closures which could actually create such a compound bus, but is of course implemented in solid-state devices. In its simplest incarnation, a single bus could be created by appropriately interconnecting multiple-bus segments to create one bus, which passes through each node. Figure 7-2 illustrates such a configuration. Active or utilized links are shown by solid lines and inactive links are shown by dashed lines. Note that there are multiple options available as to how such a bus might be constructed from the available pieces, and that if damage or a fault should disable this bus an equivalent bus could be constructed bypassing the damaged link. Figure 7-3 illustrates such an alternate configuration.

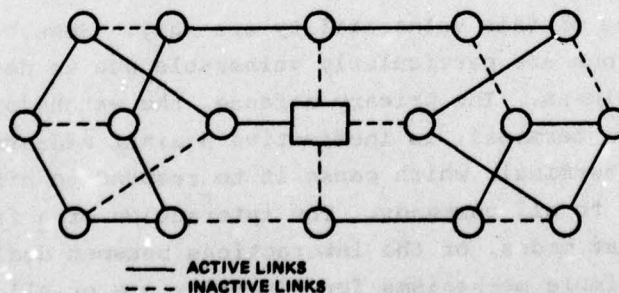


Figure 7-2. Network with virtual bus shown.



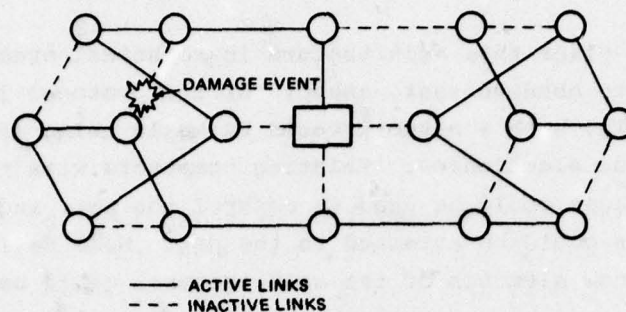


Figure 7-3. Alternate virtual bus structure.

It is from this basic ability to reconfigure the bus routing that the high-survival characteristics of the network are derived. Note, that once a bus has been created, it does indeed operate exactly as a true bus using standard bus protocols. Thus, there are no operational overheads associated with the operation of the virtual bus beyond those imposed by a standard bus and an initial setup or configuration procedure.

Damage containment and isolation of a remote terminal, which is disabling the bus, is now simple. First, each node is designed so that the interconnection circuitry provides isolation between bus segments. Electrical accidents are thereby blocked from propagating along the bus. At worst, such an accident can destroy only the isolation devices at the link terminations immediately surrounding the accident site. The logical impact of an accident, which is to disable the bus, can be overcome by reconfiguration. Note that since physical damage is confined to the immediate locale of damage, the success of reconfiguration is assured once the faulty components have been purged. Similarly a babbling remote terminal can be excised from the bus. Remote terminals can be attached at a node, or alternatively (but less desirably) along a bus segment using a 1553 stub arrangement. To excise a babbling terminal, the node to which the terminal interfaces, or the bus segment to which it is attached, can be dropped from the virtual bus. The system reconfigures around the faulty device.

The electrical constraint on numbers of terminals that can be interconnected is also eliminated. Since each node now uses active components to provide the electrical isolation between bus segments, the signaling waveform is regenerated at each node. An almost limitless number of terminals can be added without degrading the signal. This does not, of course, overcome protocol limits on the number of terminals, such as occurs in 1553.

To better place this architecture in technical perspective, it is interesting to observe that, except for the protocol limit on numbers of terminals, such a network could be built using 1553 technology for link and node electronics. Existing computers with nearly standard 1553 interfaces could be used to control the net, and any 1553 remote terminals could be attached to the net. Node devices, which are the unique new elements of the architecture, could be fairly economically fabricated by capitalizing on 1553 microcircuit components.

In addition to overcoming the three primary weaknesses of 1553, inability to interconnect a large number of terminals, damage vulnerability, and vulnerability to a babbling terminal, the network enhances 1553 performance in other ways. While these benefits are secondary and not adequate to justify a change from standard 1553 practice, they are nevertheless significant.

First, unlike 1553 buses, it is possible for the virtual bus to "Y" or branch. Since nodes are active devices, the reflections and impedance mismatches, which preclude this in a standard 1553 bus, are not relevant. Thus, a virtual bus can look like a tree, much as shown in Figure 7-4. This considerably loosens topological and routing constraints.

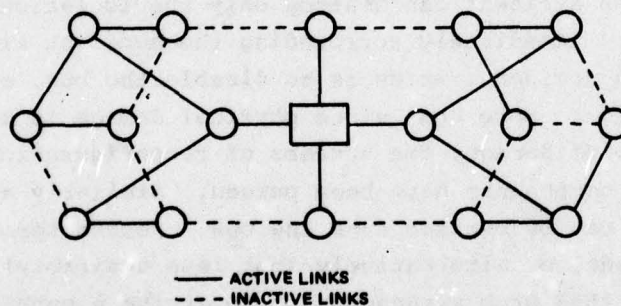


Figure 7-4. Virtual bus with "Y" constructs.

Secondly, multiple buses can be active simultaneously, and the spare or inactive links constitute a shared redundancy pool, able to repair failures in either or both buses. By using this multibus capability, it is possible to set up several buses, possibly partitioning the system according to a natural hierarchy along with dedicated point-to-point paths to link terminals with high-bandwidth requirements. Redundancy is then available inexpensively in the form of a pool of unused links. Figure 7-5 illustrates a sample configuration



with an active bus and an inactive bus, as well as a dedicated path between nodes A and B. Figure 7-6 illustrates an alternative configuration designed to overcome the local damage event which disabled node C.

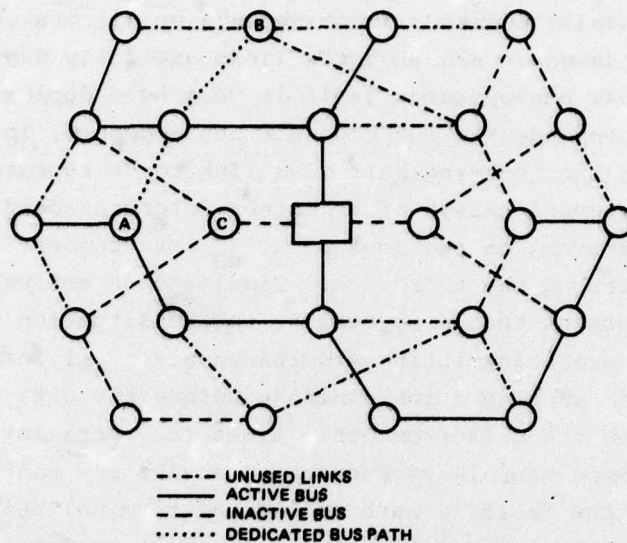


Figure 7-5. Multibus network.

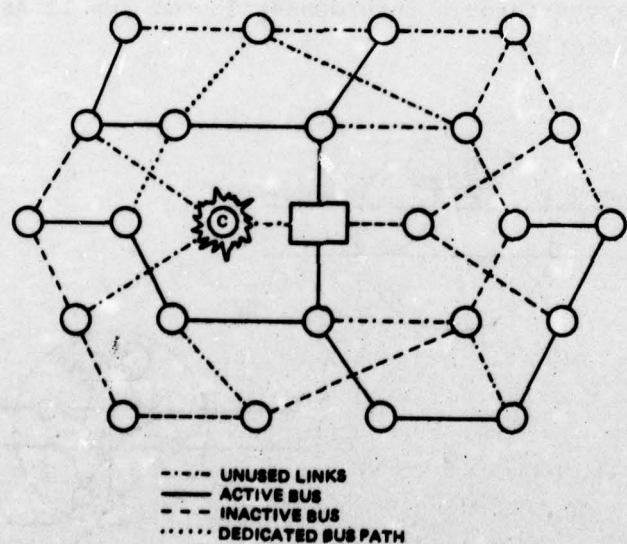


Figure 7-6. Reconfigured multibus network.

Configuration control algorithms are quite simple for maintenance of one bus, and become more complex for multiple buses of different criticalities. Configuration-control information or commands are carried to the nodes over the links from a configuration controller at one of the nodes. The links between nodes are fully duplex, and each node continually monitors incoming data on all its links for configuration messages. Although the links are fully duplex (unlike 1553), the virtual bus operates as if it were half duplex (like 1553). When a node is commanded to interconnect bus segments, it causes any data arriving on the incoming half of a link to be repeated or retransmitted on the outgoing halves of the other interconnected links. Transmissions arriving on two arms of a "Y" interconnect are combined for retransmission on the third arm. Simultaneous arrivals produce erroneous bus data on that arm, similar to the situation when two terminals on a bus are transmitting simultaneously. All incoming links are monitored for configuration commands before the data from that link are combined with the data from other links for retransmission. This assures that a node can always correctly receive any configuration commands if the bus fault is outboard of the terminal being addressed. The node immediately inboard of the fault can, therefore, receive the configuration commands necessary to disconnect the fault from the virtual bus. The configuration controller can then establish a new set of links to bypass the fault and reconnect to any nodes that were outboard of the fault. Figure 7-7 illustrates the internal interconnections for a straight-through interconnection of two links; and also for "Y" interconnection.

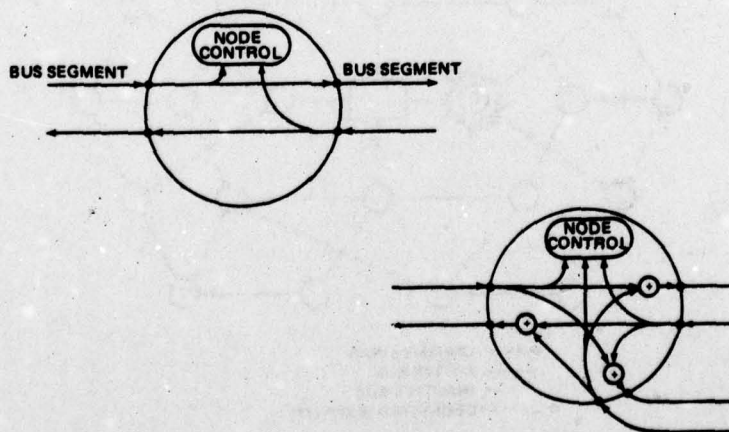


Figure 7-7. Internal node interconnections.



Figure 7-8 better illustrates the meanings of outboard and inboard. The inboard link relative to a particular bus controller is that link which is used in transmitting data to the controller, and from which data are received from the controller. Any other links interconnected with that link are outboard links. Any node Z is outboard of a particular node Y if data transmitted from node Y to Z must use an outboard link of Y. Any node Y is inboard of a particular node Z if data transmitted from node Z to Y must use the inboard link of Z. Note, that by this definition, if Z is outboard of Y, then Y is inboard of Z; however, if Y is inboard of X, X may be inboard of Y (see Figure 7-8).

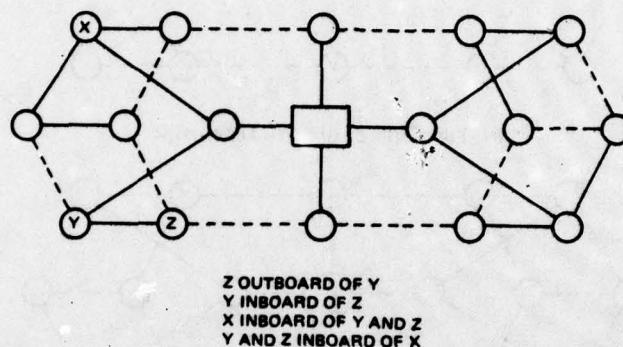
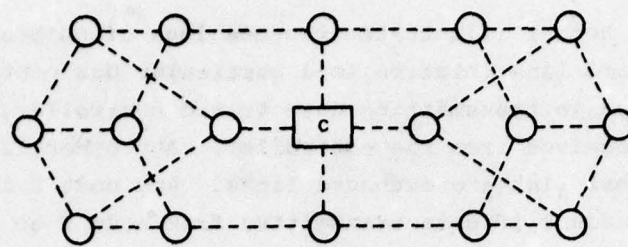
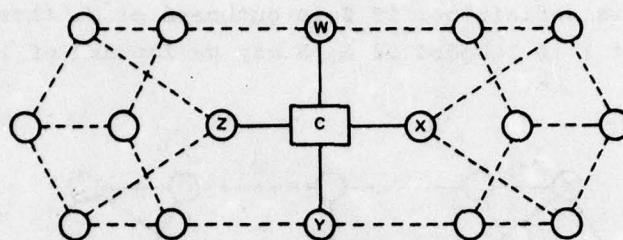


Figure 7-8. Outboard and inboard data links.

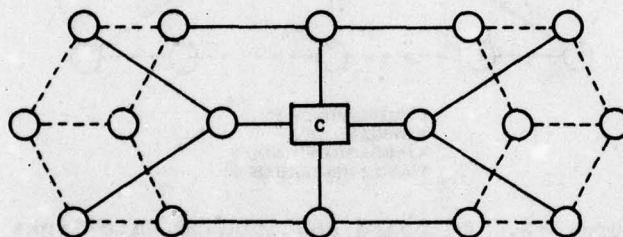
An example of the basic simplicity of the configuration algorithms can be given. Consider the case of initial configuration selection. A configuration controller at node C of the net shown in Figure 7-9(a) wishes to establish a virtual bus joining all nodes of the net. It initially activates all links radiating from node C, thereby joining nodes W, X, Y, and Z. As it establishes contact with each node, it tests the integrity of the data path, and will back off and deactivate any link activation which causes bus errors. The configuration controller then sequentially attempts activation of all the links radiating from nodes W, X, Y, and Z, which connect to nodes not already joined to the virtual bus. This procedure is repeated as new nodes are joined to the bus until all nodes have been joined or no useful links remain. This simple algorithm is guaranteed to join all nodes that have not been completely isolated by faults. Once configured, this virtual bus functions exactly as a bus and Node C could then act as the bus controller, in the sense of a 1553 bus controller, or any other node could assume the role of bus controller, or bus control could be passed dynamically from node to node. Figure 7-9(b), (c), and (d) illustrate these growth steps.



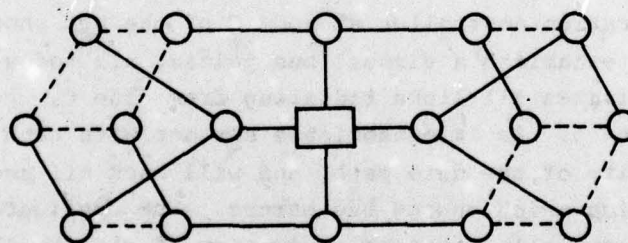
(a) UNFORMATTED NETWORK



(b) FIRST STEP OF GROWTH ALGORITHM



(c) SECOND STEP OF GROWTH ALGORITHM



(d) FINAL STEP OF GROWTH ALGORITHM

Figure 7-9. Configuration controller net.

Such an algorithm has been programmed and used to configure experimental nets. It uses less than 256 words of code for the simple 16-bit processor used as the configuration controller. The entire configuration control for maintaining, repairing, testing, and diagnosing faults for the single virtual bus case required less than 1K of code.



It is intended that the configuration controller for the network be the fault-tolerant multiprocessor, as specified in Section 3. While it is not essential that the network employ a fault-tolerant multiprocessor as configuration controller or that the multiprocessor use a network for I/O traffic, these two concepts are synergistic when combined. The network provides the multiprocessor with unequalled reliability in I/O communications, and the multiprocessor enhances the network concept by providing highly reliable configuration control.

#### 7.5 Detailed Network Design

This subsection discusses details of the network design. These details are not critical to the design concept and indeed represent only one of many implementation options. They are presented as an example and an exercise so that the technical issues might be better understood. It would be expected that any actual implementation would be further optimized for the specific situation, and might differ considerably from this proposal in detail, while still conforming to the architectural specification of Section 7.4.

##### 7.5.1 Communications Protocol

The communications protocol conforms to MIL-STD-1553B, except in the following detail.

For routine bus controller to remote terminal data traffic:

- (1) The remote terminal address field is extended from three to eight bits, and the subaddress/mode field is shortened from three to two bits.
- (2) For the shortened subaddress/mode field, only a 11 implies that the word count/mode code is to be interpreted as a mode code.
- (3) The remote terminal address 00000000 is reserved to indicate a node configuration command.

Figure 7-10 illustrates the command, status and data-word formats for this protocol. For reference purposes MIL-STD-1553B is enclosed as Appendix 7-A.

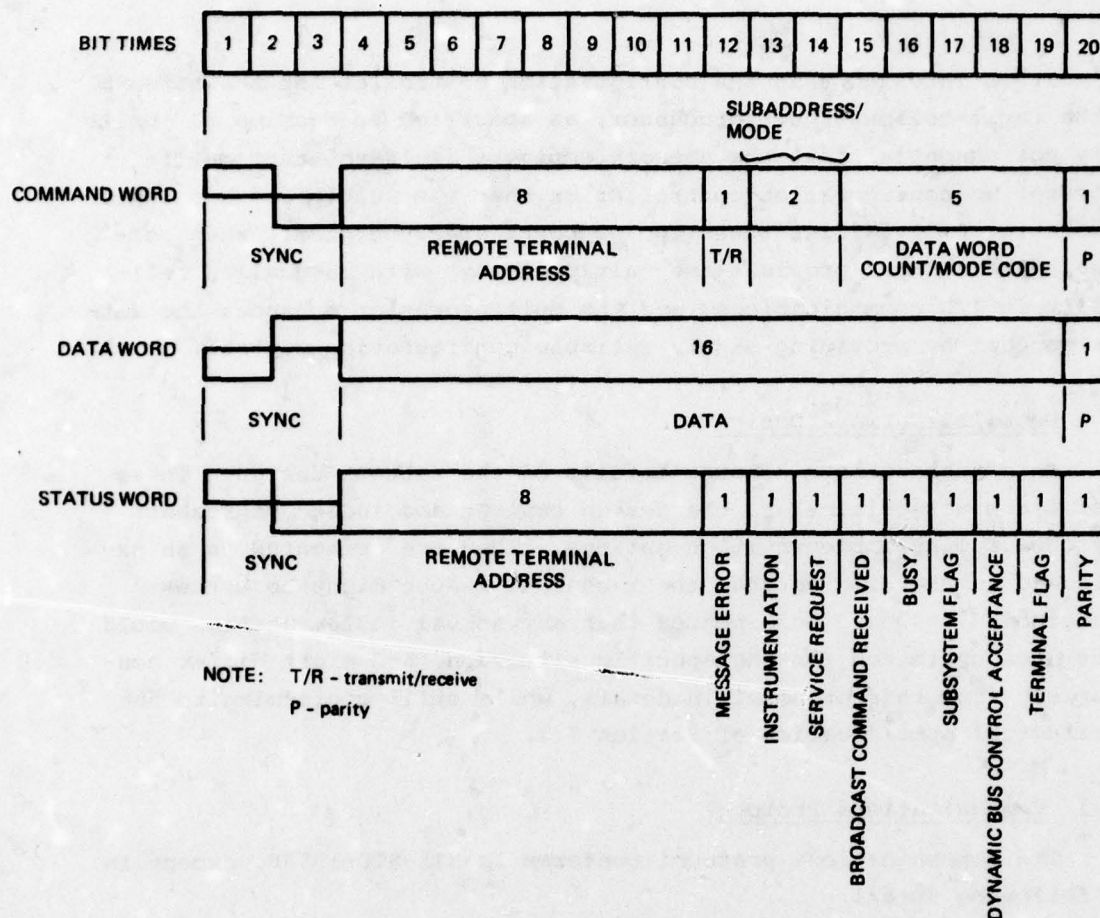


Figure 7-10. Word formats.

Node commands are differentiated from routine data traffic by appearing to be addressed to remote terminal zero. Thus, they are ignored by remote-terminal hardware. The remaining bits of the command word are redesignated as a node identifier. One data word always follows the node command word, and contains the specific node command to be executed. Any node response is identical. The first word contains a remote terminal address zero, for compatibility with routine data traffic. The second word is the node identification, which contains the requested status information. Figure 7-11 illustrates these node command and response formats.

The addressing of nodes is completely separate and distinct from the addressing of all remote terminals.



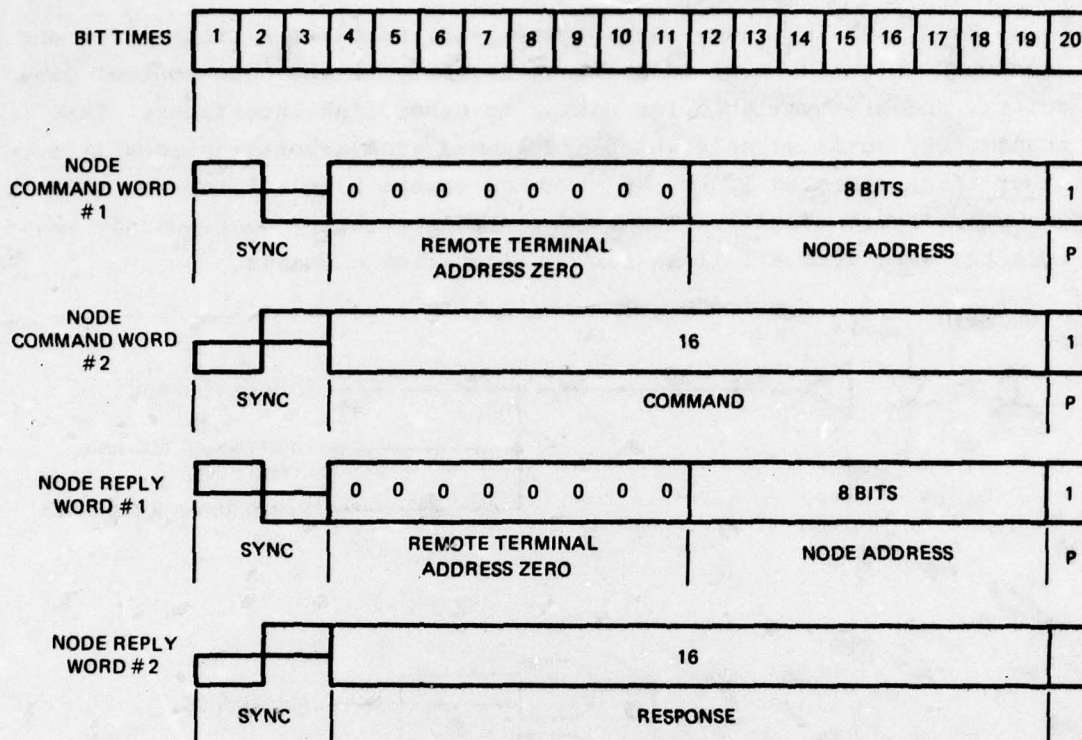


Figure 7-11. Node command/response format.

#### 7.5.2 Transmission Technology—Link Technology

At this time, it appears likely that fiber-optic technology will be ready for incorporation in 1990s networks. Since the network topology requires only point-to-point links, all the needed components exist, and facsimiles thereof have been flown. Militarized connectors, cables, and suitable transmitters and receivers are being developed under the AVIOPTICS program.

Since no power splitting is necessary, the links may operate at moderate power levels with comfortable noise margins producing a non-observable error rate. The following arrangement is suitable:

Waveguide: Medium-loss single or multiple fiber cable.

Emitters: IR light-emitting devices (LEDs).

Detectors: PIN Photodiodes.

Modulation Technique: Pulse-position modulation; moderate peak power and pulse width.

Multiport Couplers: None.

Each link interface (see Figure 7-12) contains a transmitter and a receiver. Signals received are sent directly to the node control circuitry, and are available for gating to other link interfaces. The transmitter sends signals which are summed from several sources (i.e., other interconnected links, an attached remote terminal, or the node control circuit itself). The node control circuitry continuously monitors the data from all links for configuration commands.

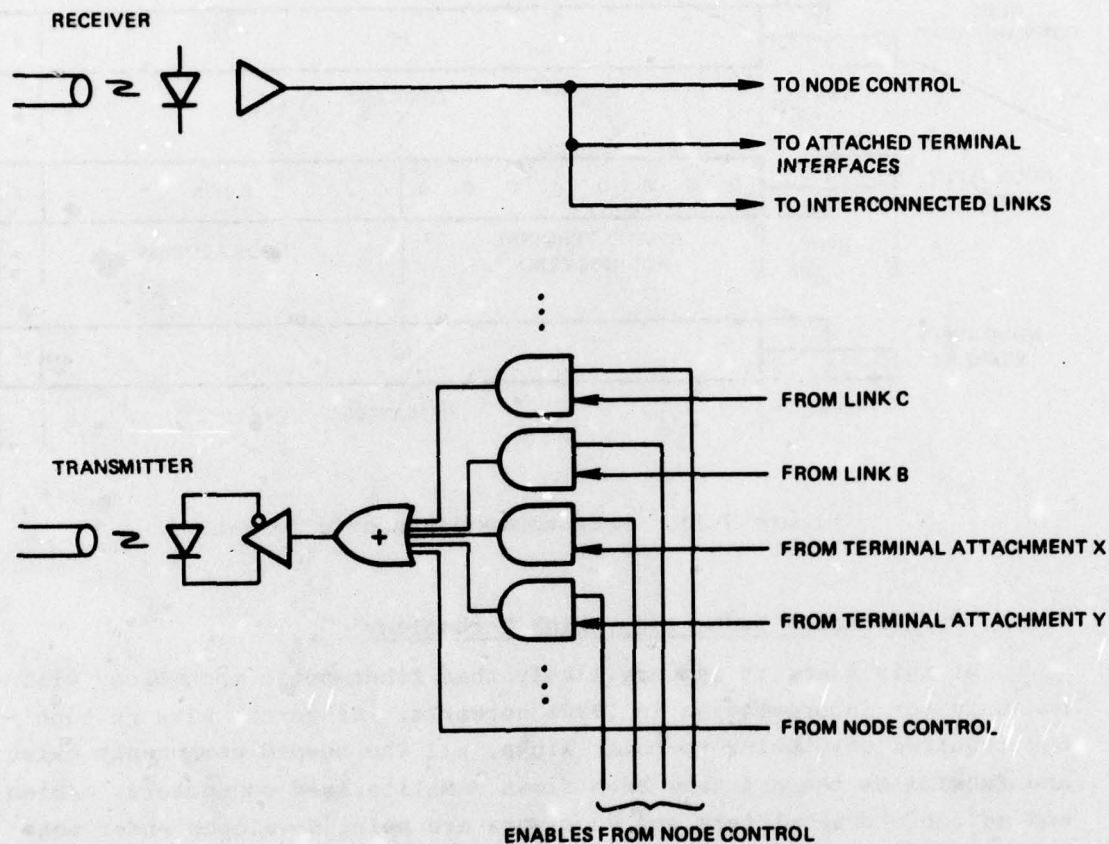


Figure 7-12. Link termination circuitry.

Pulse-position modulation is used to encode the impressed Manchester encoding onto the data stream, and to encode the data. Figure 7-13 illustrates the encoding format used. This corresponds exactly to the pulse-position modulation technique being used by Marconi Avionics and IBM in their experimental fiber-optic 1553 data buses. The transmitter circuit is designed to regenerate the pulse-wave shape so that successive bufferings will not degrade the signal. Note that un-



like the IBM effort and like the Marconi implementation, only point-to-point transmission is required. Thus, it could be expected that multi-megabit data rates could be easily supported if desirable.

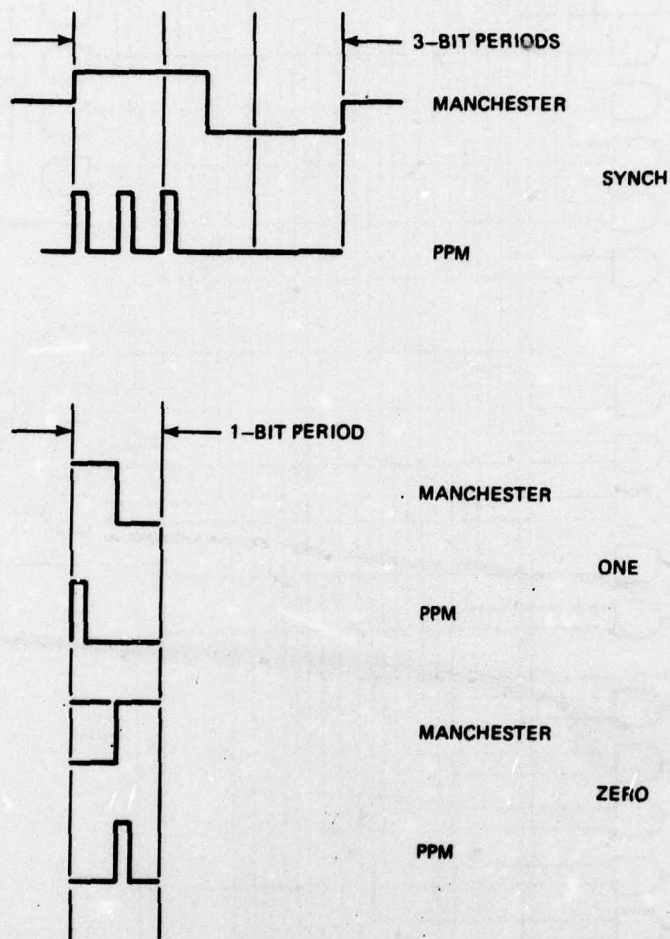


Figure 7-13. Pulse-position modulation (PPM).

### 7.5.3 Node Construction

The node is constructed as is shown in Figure 7-14. The node control can listen to all links directly and can respond to configuration commands regardless of the internal node interconnections. Enabling signals are provided by the node control, which interconnects the various serial links, or which can be used to interconnect to the remote terminal. The remote terminal(s) or subscribers attached to the node

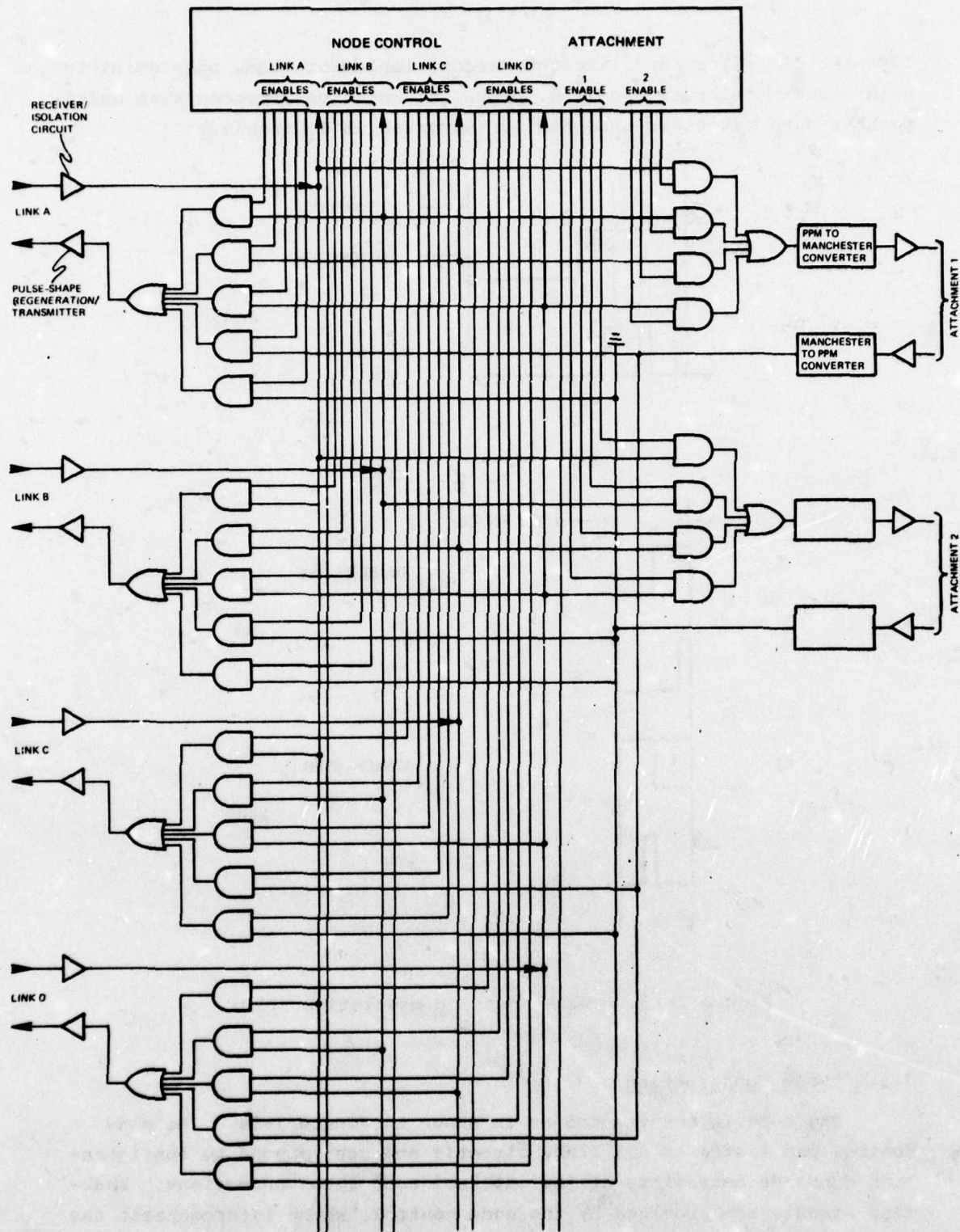


Figure 7-14. Node functional organization.



communicate with the net using the standard net protocols. The connection may be either fiber-optic or electrical. An electrical connection conforms to 1553 bus electrical specifications. There are two remote-terminal attachments on a node, and they may be interconnected to the same virtual bus through the node or to separate virtual buses passing through the node. It is, of course, possible for a virtual bus to pass through the node without being interconnected to either remote-terminal attachment.

An optional node attachment adapter will convert the standard serial protocol format to a parallel format providing address recognition, mode command execution, and limited buffering of data. Terminal attachments to the node are discussed in greater detail in the next section.

#### 7.5.4 Terminal Attachment to the Network

All elements (i.e., sensors, actuators, displays, controls, etc.) communicate over the network through terminals attached to the network nodes. It is envisioned that the network and its nodes will serve as a more-or-less-permanently installed facility in the aircraft. Terminals, on the other hand, may change as mission needs or technology advances dictate.

The primary terminal interface is a direct attachment through one of the terminal attachment parts of the node. Attachment at this point requires that the remote terminal conform to the serial bus protocol used by the network. Design of this protocol has been such as to maximize the similarities with MIL-STD-1553B protocol, and it is expected that MIL-STD-1553B devices could be attached directly to one of these ports with only minor changes. New devices could be built to this standard using available 1553 microcircuit technology. Note that since the primary attachment is still a bus protocol, it is possible to attach several remote terminals to the same port of a node in parallel. This is not necessarily recommended, however, because of the implications of a node failure which could lead to loss of all attached terminals.

A family of secondary interfaces is also proposed for converting from the primary standard to secondary standards. These will include a parallel interface, a pure 1553 interface, and various simple full-function interfaces for discrete sensing and actuation, A/D and D/A conversion, power control, etc. Since the attachment point is still a

bus, several of these secondary interfaces may also be attached to the same port. Each secondary interface must have a unique terminal address and will perform the necessary address decoding and recognition.

Note that two levels of protection are provided against a babbling terminal. First, the node can be instructed to disconnect that terminal's attachment point. Secondly, surrounding nodes can disconnect from the node itself and route the bus entirely around the faulty node/terminal.

#### 7.5.5 Mechanical Design

Nodes may be mounted at various points in avionics bays, installed within certain equipment, and distributed throughout the aircraft in potentially hostile environments. It will be necessary to provide EMI shielding, particularly of optical receivers. While fibers are apparently immune to electromagnetic problems, the electronics by itself is most certainly not. Since nodes will be installed in large numbers, they must possess modest size, weight, and power requirements.

#### Node Components

Since nodes will be produced in large quantities, specialized integrated circuits may be assumed for the major functions. These would be interconnected in a hybrid circuit installed in an enclosure, appropriate to the node's intended location in the aircraft, equipped with mounting means and connectors.

The major node components, and their sizes, are as listed:

- |                                 |   |
|---------------------------------|---|
| (1) Node Control (central)      | 1 Digital LSI chip  |
| (2) Each Attachment Port        | 1 Bipolar chip  |
| (3) Crystal Oscillator          | { 1 Linear IC<br>1 Quartz Crystal<br>6 Passive Components |
| (4) Link Interface (4 required) |   |
| (a) Receiver:                   |   |
| Detector                        | 1 Diode Chip  |
| Amplifier                       | { 1 Linear Chip<br>6 Passive Components                   |
| Distributed Dedicated Decode    |   |
| Node Control/Link               | 1 LSI Chip  |



(b) Transmitter:

Gating/Modulation	{1 Bipolar Chip, 6 Passive Components
Emitter Driver	{1 Transistor, 4 Passive Components
Emitter	1 LED

(5) Power Conditioner

3 Watts (maximum) at 2 W/in.<sup>3</sup>

If packaged in hybrid form, we expect the connectors to consume more space than the node components. The node size should not exceed 6 cubic inches, including connectors and enclosures.

#### 7.6 Summary

The integrated system demands a great deal from a data-communications medium in the way of flexibility and rigorous dependability. The 1553 standard as currently implemented cannot meet these requirements. Except for the 1553 protocol limitations on the numbers of terminals which can be interconnected, a networking alternative exists that could meet most of the dependability and flexibility requirements while still maintaining functional compatibility with 1553 devices. The specific proposal of Section 7.5 details a minimum modification to 1553 protocol to meet the remaining terminal numbers limitation. Such an approach represents a minimum-risk development of the network concept, and draws heavily from the 1553 technology base.

The exclusive use of point-to-point links in the network is particularly advantageous with regard to fiber optics. Such a link should be well within the current technology and the AVIOPTICS program might be encouraged to seek an optimized link design for such an application.

Networking and, indeed, ultra-high reliability data multiplexing are new to avionics. A good deal of exploratory work remains to be done. Questions which should be studied further deal with optimum protocol selection, control algorithms for multiple configuration optimization, and topological implications of various network layouts.

## SECTION 8

### SOFTWARE AND RESOURCE SIZING

The purpose of this section is to outline the core-avionics software functions needed for an advanced Naval aircraft, and to assemble throughput and memory estimates for that software. These estimates are necessary to allow preliminary sizing of the avionics computation capability and to determine a reasonable distribution of processing resources throughout the avionics system.

The avionics-system architecture is described in detail in related sections. For purposes of this discussion it suffices to define the system as follows:

- (1) A high-level fault-tolerant computer complex (FTMP).
- (2) Some number of local computers, each of which bears a unique relationship with one or a small group of subsystems. (Such computers may be physically embedded in the related subsystem).
- (3) A communications network to which all computers are attached—each computer being attached at a number of nodes of the network.

It is assumed that the network is configured as a single multiplexed communications channel (a bus), which is controlled from the fault-tolerant computer. The configuration of paths from node to node may change with time, but at any one time there exists only a single channel. This is not to preclude the use of multiple simultaneous network paths, but to provide a baseline assumption for the analysis of software distribution. While the introduction of multiple paths increases network bandwidth, it also introduces operating complexity. This burden may not be necessary if single channel bandwidth is adequate.

Distributed computation offers a number of technical and management advantages; however, it is not a magical solution to the throughput limitations presented by a single computer, or to the complexities inherent in multiprogramming. If some processing can be relegated to



local or embedded computers then this will, of course, relieve some of the burden on the computer. The extent to which a particular software function can or should be distributed depends upon the following:

- (1) The information interfaces of a particular software function.
- (2) The bandwidth of the network communications channel.
- (3) The processing-throughput capacity of the fault-tolerant computer.
- (4) The redundancy-management requirements for the related subsystem.

Clearly, some functions, such as network reconfiguration control, must be centralized by their very nature. Other functions, such as flight control, interface with a large variety of sensor and effector subsystems, and naturally belong in a central location. Conversely, items such as sensor compensation for a strapdown inertial reference package should be done locally to minimize the data traffic over the network. Finally there exist functions that could be reasonably done centrally or locally. Their placement dictates (or is dictated by) network bandwidth, central computer throughput, and redundancy-management requirements.

The next section gives a brief description and breakdown of the core avionics functions; that is, those functions related to flying the aircraft, but which are not mission-specific. These functions have been generalized for an advanced Naval aircraft with leanings towards a V/STOL. Table 8-1\* summarizes memory and throughput estimates for the given breakdown. These were derived primarily from examination of similar functions for existing aircraft and spacecraft. In some cases, such as landing guidance for V/STOL mode, specific algorithms were analyzed.

It is not the main purpose of this discussion to assign all functions to either central or local computation, although some functions will obviously fall into one category or the other. The intent is primarily to estimate throughput requirements for each function, so that these data may assist the detailed system architecture development.

---

\*Table 8-1 is found at the end of this section due to its length.

As a side note, it appears that memory requirements will have much less of an impact on system architecture (of the type considered) than processing-throughput and network-bandwidth requirements.

### 8.1 Core-Avionics Functions

This section contains brief descriptions of the core-avionics functions. The actual throughput and memory estimates for these functions are contained in similarly numbered sections of Table 8-1. Where possible, these figures are derived from the program statistics of similar functions in other aerospace systems. In other cases, the figures are estimated from assumed algorithms or assumed volume of data processing. The particular systems referenced include the following:

- (1) The NASA Space Shuttle primary GN&C system, the Backup Flight System (BFS), and the approach and landing test Backup Flight Control System (BFCS). During atmospheric flight this vehicle behaves like a conventional airplane. During orbital operations the flight-control system has characteristics that correspond to a V/STOL hover mode. Additionally, the cockpit contains multifunction CRT display systems and keyboard data-entry mechanisms.
- (2) Air Force Interim Upper Stage (IUS). This employs a five-gyro five-accelerometer strapdown inertial-navigation system, which is similar to the instrument complement proposed for V/STOL.
- (3) Boeing study for a digital-flight-control system for a B-504 ASW aircraft (NADC Technical Report 76134-20).
- (4) F-8 Digital Fly-By-Wire control system. An advanced digital-flight-control system for a fighter aircraft (a CSDL development).
- (5) Draper Report R-1151. Integrated navigation using GPS, JTIDS and inertial systems.
- (6) Navy's Advanced Integrated Display System (AIDS).
- (7) Navy's Advanced Aircraft Electrical System (AAES).

#### 8.1.1 Flight Control

This encompasses vehicle stabilization, attitude control, and thrust control. Figures for the basic control algorithm are taken



from the Boeing study, because it was directed toward an ASW aircraft. The F-8 figures are about 50 percent higher in both throughput and memory requirements for a somewhat more demanding fighter application. The two sets of figures appear consistent.

Flight director computations are from the Boeing study.

Attitude control for hovering mode may be implemented via main-engine pitch, and/or reaction control jets driven by the main-engine compressors. Figures for such jet-selection logic are taken from Shuttle entry-flight-control functions, as these most closely parallel the V/STOL problem.

#### 8.1.2 Guidance

This refers to automatic shipboard-landing guidance—transition from horizontal to vertical flight and landing. Figures are estimated from developed algorithms.

Cruise control (i.e., conventional airplane autopilot function) is actually part of flight control.

#### 8.1.3 Navigation

Basic navigation includes propagation of vehicle state vector via inertial data, GPS satellite data and JTIDS time-of-arrival data. The navigation employs a Kalman filter that utilizes the aforementioned sources. These figures are from CSDL Report R-1151.

#### 8.1.4 Crew Interface

This deals with cockpit displays and avionics system mode controls.

##### 8.1.4.1 Displays

Displays are considered to be multipurpose CRTs as described in relevant AIDS documentation. Scaling and other preparation of FTMP data for display are estimated based on a total of about 100 values displayed at any one time on the five main pilot display units. Memory accounts for about 1000 displayable values.

CRT display formats represent the fixed portion of displays and the mapping of variable data to specific locations on the CRT. Actual computations are hardware-assisted (such as symbol rotation, etc.), or embedded at the CRT. The AIDS raster signal generator is indicative of this sort of hardware. Storage requirements for formats are from Shuttle display format figures.

CRT refresh is a hardware function.

Special moding indicators and caution and warning annunciation are estimated based on Shuttle experience.

#### 8.1.4.2 Keyboard and Moding Commands

The scanning of keyboard and moding discretes for state changes is fairly simple, but high-rate (10 hertz). Figures are estimated from the AIDS 30-key integrated control panel, including redundancy management on the switches.

Processing of keyboard commands is done on an as-required basis (i.e., when a message has been entered at a keyboard). These computations can be fairly low priority and still provide adequate response to the crew. Memory requirements come from the Shuttle keyboard-processing programs.

#### 8.1.5 Communications

Data-communications services are provided by JTIDS, a complex radio-communications network that allows air-going and sea-going vehicles to swap navigational and tactical information. JTIDS baseline information has been taken from CSDL Report R-1151. The relative navigation portions have been extracted from the JTIDS baseline, and are included in the general navigation function. TACAN processing has also been extracted and treated as a separate subsystem. What remains is tactical communication that has almost no interaction with the guidance, navigation, and control (GN&C) operations.

#### 8.1.6 Subsystem Processing

Subsystems include hardware items connected to the data network; they may be simple sensors or complex devices. The real-world information provided by, or commands sent to, these items requires some amount of processing. Such processing may be done locally by a computer associated directly with the item, or centrally in the FTMP.

##### 8.1.6.1 GN&C Subsystems Where Inputs Require Simple Processing

These subsystems include sensors, actuators, and radio equipment related to Guidance, Navigation, and Control functions.



This group is comprised of manual pilot guidance commands and various actuator position feedback sensors. Computations include scaling, biasing, and formatting raw data into floating point. Redundancy management (e.g., mid-value selection) is included. Figures for Shuttle functions are taken from Shuttle software. Figures for non-Shuttle functions are derived from Shuttle functions that have similar computation characteristics.

#### 8.1.6.2 GN&C Subsystems Where Inputs Require Complex Processing

Air data figures are derived from the Shuttle backup flight system.

The rate gyro and accelerometer complement for flight control consists of four skewed instruments of each type. Processing includes hard error detection and transformation of rates into vehicle body coordinates, and is functionally similar to corresponding operations on the navigation-gyro cluster used by IUS. Figures are derived from IUS software.

The inertial navigation cluster consists of four gyros and four accelerometers in a configuration similar to the IUS cluster. IUS software figures are used.

GPS figures are taken from CSDL Report R-1151. Navigation and the navigation filter function have been incorporated under central navigation (see Section 8.1.3).

TACAN figures are taken from CSDL Report R-1151.

Doppler radar and short-range landing system figures are estimated based on the computation associated with these functions.

#### 8.1.6.3 Equipment for U.S. Continental Airspace

This is required primarily for operation out of U.S. civilian airports. Radar altitude computations are taken from Shuttle BFCS figures. Other figures are estimated.

#### 8.1.6.4 Analog and Discrete Outputs

This group includes scaling, biasing, and the distribution to redundant actuators of commands such as aerosurface position, etc. Figures are derived from similar Shuttle functions.

#### 8.1.6.5 Vehicle-Management Subsystems

These computations monitor and control the state of such things as fuel, environmental systems, and ice buildup. The figures for the hydraulic system monitor are from the Shuttle BFS. This monitor runs at a high rate, commensurate with that for aerosurface control. Electrical system monitors are derived using AAES concepts. Others are rough estimates.

#### 8.1.7 FTMP Operating System

The operating system has two fundamental functions: control of process execution, memory management, and fault recovery within the FTMP; and control of the network configuration, including the FTMP interface with the network and network redundancy management. Additionally, system fault annunciation falls under the umbrella of the operating system, as does the run-time library (although the execution time of these library functions is generally charged to the caller).

The memory estimates given are derived from Shuttle software. Operating system throughput requirements are difficult to define at this point; however, the total overhead due to the operating system and to the characteristics of multiprocessor operation is estimated to be in the range of 70 to 100 percent of the applications code throughput requirements. In other words, about 40 to 50 percent of computer operations will be devoted to overhead.

#### 8.1.8 Data Base

The data base is estimated to be approximately 25 percent of program size, based on similar proportions experienced on the Shuttle, IUS, etc.

#### 8.2 Summary

The detailed processing throughput and memory figures in Table 8-1 have been combined, in the following chart, into several broad categories.



### Summary of Processing Throughput and Memory Requirements

<u>Function</u>	<u>KOPS</u>	<u>Memory (16-Bit Words)</u>
JTIDS	345	37300
GPS	100	22350
Inertial Navigation Instruments	81	7360
FTMP Applications	282	55010
FTMP Operating Systems and Overhead	300	25400
FTMP Data Base	-	20000
Display Formats at CRT	-	20000

- (1) JTIDS communication processing represents computations relegated to a local JTIDS processor.
- (2) GPS front-end data processing is also assumed to be done locally.
- (3) Inertial Navigation System (INS) functions related to instrument compensation, fault detection, and base motion isolation are best done locally. Alignment is not included in this KOP figure.
- (4) The remaining core applications (excluding local display operations) total less than 300 KOPS. Although some of these functions may be done locally for purposes of off-loading network bandwidth, consider as a first cut that they are all done in the FTMP. The operating system throughput requirements for the FTMP can be conservatively estimated to be equal to that required for the applications functions, or about 300 KOPS.
- (5) CRT local display-format memory requirements and the FTMP data base have also been noted above.

Even if the central computer is responsible for some functions that might well be done locally (as suggested in Table 8-1), the 600 KOPS requirement seems unlikely to strain the FTMP, which is proposed as a 2-mega-op computer. Memory requirements are well within the state-of-the-art for semiconductor memories. In the event that the core avionics function should increase by a factor of 2 or 3, the evolving LSI technologies should be easily able to accommodate this increase within the proposed system architecture.

Table 8-1. Processing throughput and memory requirements.

Section	Function	KOPS	Memory*	Source	Execution**
8.1.1	FLIGHT CONTROL				
	Mission-Critical FCS (30 Hz)	21.9	3000	Boeing	C
	Flight Director Comps (30 Hz)	2.2	850	Boeing	C
	Attitude Control Jet-Selection Logic	19.0	500	Shuttle	C
8.1.2	GUIDANCE (8 Hz) (Horizontal Flight Transition to Landing)	20.0	5000	Estimate	C
8.1.3	NAVIGATION (10 Hz)				
	Basic Nav (Combined Radio & Inertial)	32.0	2150	R-1151	C
	Nav Filter	60.0	8500	R-1151	C
	Processing for Displays	2.0	200	Estimate	C
8.1.4	CREW INTERFACE				
	DISPLAYS (Using Basic AIDS Elements)				
	Display Data Preparation	10.0	5000	Estimate	C
	Display Formats (at CRTs)	None	20000	Shuttle	L
	Moding Displays and C&W	1.0	500	Estimate	C/L
	KEYBOARD AND MODING COMMANDS				
	Lexical Processing	(low	1000	Shuttle	C
	Semantic Processing	prior-ity	6000	Shuttle	C
	Error Detection	backgnd jobs)	3000	Shuttle	C
	Keyboard Scan and Talkback	10.0	600	Aids/Estimate	L

\* Memory given in 16-bit words.

\*\*Likely location for execution of the function.

C = central computer (FTMP)

L = local processor



Table 8-1. Processing throughput and memory requirements (Cont).

Section	Function	KOPS	Memory*	Source	Execution**
8.1.5	COMMUNICATIONS (JTIDS)				
	Exec and Services	42.2	3300	R-1151	L
	Network Processing	108.9	8750	R-1151	L
	Message Processing	75.3	3750	R-1151	L
	Data-Base Processing	33.5	750	R-1151	L
	Display Processing	50.2	3500	R-1151	L
	Subscriber Processing	20.9	1100	R-1151	L
	Recording	12.6	500	R-1151	L
	Data Base	None	15700	R-1151	L
8.1.6	SUBSYSTEM PROCESSING				
8.1.6.1	NAVIGATION AND CONTROL SUBSYSTEMS WHERE INPUTS REQUIRE SIMPLE PROCESSING				
	PILOT INPUT COMMANDS (30 Hz)				
	Sticks	8.7	500	Shuttle	L
	Throttle	1.2	80	Shuttle	L
	Rudder	0.7	50	Shuttle	L
	Fan/Engine Vectoring	0.6	40	Shuttle	L
	Engine Blade Pitch	0.6	40	Shuttle	L
	ACTUATOR POSITION FEEDBACKS (30 Hz)				
	Rudder	0.6	40	Shuttle	L
	Elevator	1.2	80	Shuttle	L
	Ailerons	1.2	80	Shuttle	L
	Flaps	1.2	80	Shuttle	L
	ENGINE FEEDBACKS (2 Engines) (30 Hz)				
	Throttling	1.2	80	Shuttle	L
	Vectoring	1.2	80	Shuttle	L
	Blade Pitch	1.2	80	Shuttle	L
	Temperature	1.2	80	Shuttle	L
	TRIM, MODING, AND MISC PANEL SWITCHES	24.0	1600	Shuttle	L

\* Memory given in 16-bit words.

\*\*Likely location for execution of the function.

C = central computer (FTMP)

L = local processor

Table 8-1. Processing throughput and memory requirements (Cont).

Section	Function	KOPS	Memory*	Source	Execution**
8.1.6.2	GN&C INPUTS REQUIRING COMPLEX PROCESSING				
	AIR DATA (1 Hz)				
	Formatting Input Parameters	0.3	125	Shuttle	L
	Redundancy Management	0.8	375	Shuttle	L
	Air Data Equations	2.0	1000	Shuttle	L
	RATE GYROS (5)	7.0	800	IUS	L
	ACCELEROMETERS (5)	7.0	800	IUS	L
	INERTIAL NAVIGATION INSTRUMENTS				
	Sensor Compensation	45.5	3300	IUS	L
	Base Motion Isolation	13.0	560	IUS	L
	Fault Detection and Isolation	22.4	1900	IUS	L
	Alignment	4.8	1600	IUS	C/L
	GPS DATA PROCESSING				
	Executive and Services (Library)	62.0	5600	R-1151	L
	Satellite Selection	12.5	2975	R-1151	L
	Receiver Control	12.5	3725	R-1151	L
	Satellite Data Formatting	12.5	2125	R-1151	L
	Init. Calib. and Misc.	NEG	2800	R-1151	L
	Operator Interface	NEG	375	R-1151	L
	Data Base	None	4750	R-1151	L
	TACAN	25.1	2000	R-1151	L
	DOPPLER RADAR	1.0	200	Estimate	L
	SHORT-RANGE LANDING SYSTEM	5.0	1000	Estimate	L

\*Memory given in 16-bit words.

\*\*Likely location for execution of the function.

C = central computer (FTMP)

L = local processor



Table 8-1. Processing throughput and memory requirements (Cont).

Section	Function	KOPS	Memory*	Source	Execution**
8.1.6.3	EQUIPMENT FOR U.S. CONTINENTAL AIRSPACE				
	RADAR ALTIMETER AND DISPLAY CALCULATIONS (10 Hz)	0.5	100	Shuttle/BFCS	C
	Automatic Direction Finder (ADF)	NEG	NEG	Estimate	L
	MLS (Data Formatting for Nav.)	0.5	100	Estimate	L
	ATC Transponder	NEG	NEG	Estimate	L
8.1.6.4	ANALOG AND DISCRETE OUTPUTS				
	SURFACE ACTUATOR COMMANDS				
	Ailerons	0.6	50	Shuttle	L
	Rudder	0.6	50	Shuttle	L
	Elevator	0.6	50	Shuttle	L
	Flaps	NEG	NEG	Shuttle	
	LANDING GEAR	NEG	NEG	Shuttle	
	ENGINE COMMANDS (2 ENGINES)				
	Throttles	1.2	100	Shuttle	L
	Blade Pitch	1.2	100	Shuttle	L
	Vectoring	1.2	100	Shuttle	L
	Startup	NEG	NEG	Shuttle	L
	REACTION JETS (FORMATTING)	3.6	1200	Shuttle	C
8.1.6.5	VEHICLE MANAGEMENT SUBSYSTEMS				
	MONITORS				
	Fuel	NEG	100	Estimate	C/L
	Electrical System (1 Hz)	7.5	3500	Estimate/AAES	C/L
	Environmental	NEG	200	Estimate	C/L
	Hydraulics (25 Hz)	7.5	500	Shuttle	C/L
	Icing	NEG	200	Estimate	C/L

\* Memory given in 16-bit words.

\*\*Likely location for execution of the function.

C = central computer (FTMP)

L = local processor

Table 8-1. Processing throughput and memory requirements (Cont).

Section	Function	KOPS	Memory*	Source	Execution**
8.1.6.5	VEHICLE MANAGEMENT SUBSYSTEMS (Cont).				
	CONTROLS				
	Fuel	NEG	100	Estimate	C/L
	Electrical System	1.0	1500	Estimate/AAES	C/L
	Environmental	NEG	200	Estimate	C/L
	Hydraulics	1.0	250	Estimate	C/L
	Icing	NEG	200	Estimate	C/L
	PROCESSING FOR DISPLAYS	1.0	200	Estimate	C/L
8.1.7	FTMP OPERATING SYSTEM	†			
	Process Queue Operations		1000	Shuttle	C
	Timer Queue Operations		1000	Shuttle	C
	Event Queue Operations		400	Shuttle	C
	Data Tables for Queues		5000	Shuttle	C
	Memory Management		2000	Estimate	C
	Startup		3000	Shuttle	C
	Input/Output Control		3500	Shuttle	C
	FTMP Reconfiguration Control		1000	Estimate	C
	Network Reconfiguration Control		3000	Estimate	C
	Fault Annunciation		1000	Shuttle	C
	Error Interrupt Handling		500	Shuttle	C
	Runtime Library		4000	Shuttle	C
8.1.8	FTMP DATA BASE	None	††	Estimate	C

\* Memory given in 16-bit words.

\*\*Likely location for execution of the function.

C = central computer (FTMP)

L = local processor

† See Section 8.1.7 roughly equivalent to application throughput (i.e., 300 KOPS).

†† Approximately 25 percent of total program size.



## SECTION 9

### POWER DISTRIBUTION

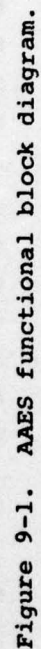
#### 9.1 Introduction

The requirements for a power source to supply the integrated fault-tolerant avionics system were investigated. Stated ideally, this requirement is for transient free power with a probability of survival which is consistent with flight-critical operation of the avionics system.

This requirement implies multiple sources of power distributed to each critical load, and adequate filtering to eliminate the transients which are common to aircraft electrical systems. The power system should be a network with several independent generators, power controls, and distribution cables which can tolerate multiple failures without loss of power at the critical system elements. The Advanced Aircraft Electrical System, currently under development by the Navy, will be examined in the context of these requirements.

#### 9.2 Advanced Aircraft Electrical System (AAES)

The AAES introduces the concept of aircraft-load management and the resultant benefits of reduced weight, higher reliability, and reduced cost. The referenced articles describe the various components and how the components are interconnected (see Figure 9-1) to provide a Load Management Center. The technologies being developed have characteristics sufficiently general so that they are adaptable to power networking required by fault-tolerant electronics. However, to critically assess the application of AAES requires a study of the detailed implementation of the concepts for an integrated fault-tolerant avionic system. Equipment location and dispersement, packaging, provision for field maintenance, and the selection and specification of semiconductor components will have a significant impact on the design of the ultimate system and its resulting performance.





Present-day aircraft are not as critically dependent on electrical power as the future aircraft that are being proposed. If future aircraft are going to use the emerging electronic technologies to perform flight-critical operations, then the aircraft power systems must also be updated to meet the new level of criticality.

The detailed design of the AAES electronic modules must take careful account of redundancy and redundancy management. For example, Figure 9-1 shows two redundant buses being driven from the same cable control unit (CCU) module. This is potentially a problem because a single failure in the CCU module could bring down both buses.

Fault-tolerant computers are designed to be powered from three or more power supplies, each power supply receiving raw power from a separate power source (see Figure 9-2). The output from the power conditioners would be silicone control rectifier or diode coupled to the internal power lines inside each module. Usually, one of these raw power sources is the aircraft battery power. The diode coupling tolerates loss of power on all power lines but one.

The AAES concept for load management requires a power controller (PC) for each load. In the case of a fault-tolerant computer and other flight-critical electronics, the use of a nonredundant PC and demultiplexer (DMUX) as indicated in Figure 9-1 would be extremely risky. Therefore, a higher level of redundancy must be provided in AAES if load management is to be permitted for flight-critical subsystems. For example, as indicated in Figure 9-2, the fault-tolerant computer would need at least three power conditioners, all independently controlled. In addition, special considerations need to be given to the power distribution to tolerate battle damage.

Assuming that flight-critical requirements are levied on the AAES, the proposed FTMP and data network would be an excellent candidate for the AAES control system. Simple dual or even triple redundancy is probably not sufficient. The flight-critical parts of AAES should probably be operational after two failures, and safe after the third. Our view of the present status of AAES is summarized in the following sections.

#### 9.2.1 Summary of AAES Features

AAES proposes to convert the present 115-volt ac, 28-volt dc power system currently used in military and commercial aircraft to a power

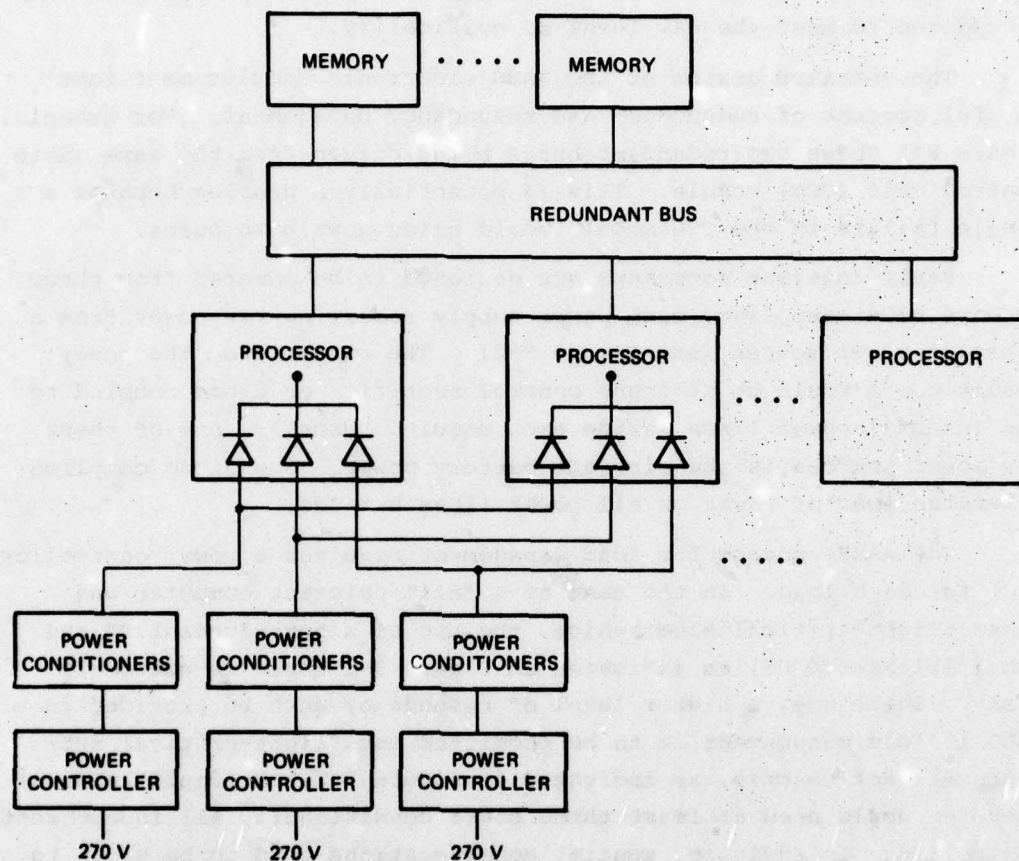


Figure 9-2. Fault-tolerant power distribution.

system which incorporates new power generators and new methods of power distribution. This system can reap a number of advantages by the utilization of advanced technologies. Some of the potential advantages are as follows:

- (1) Reduced aircraft weight.
- (2) Increased reliability.
- (3) Reduced electromagnetic interference.
- (4) Better utilization of aircraft power through power management.



- (5) Standardization of aircraft power modules.
- (6) Reduced aircraft wiring.
- (7) Built-in test and diagnostic capability.
- (8) Amenable to fault-tolerant techniques.

The AAES plans to gain these advantages by introducing a power system with the following three major components (see Figure 9-1).

- (1) Power Generating System (PGS), which introduces the new high-voltage dc generators.
- (2) Solid-state electric logic (SOSTEL), which has solid-state switches that can be activated by the SOSTEL processor for load control and power management.
- (3) General-Purpose Multiplex System (GPMS), which can provide redundant communication between the SOSTEL modules.

All components of the system have built-in test (BIT) capability. With the AAES building blocks (PGS, SOSTEL, and GPMS) a load management center (LMC) concept can be implemented, which can provide the following advantages:

- (1) Better utilization of power—The SOSTEL processor can turn off equipment not required during selected mission modes.
- (2) Graceful degradation—The SOSTEL can turn off power on a priority basis when the PGS cannot supply the total demand power.
- (3) Transient reduction—The SOSTEL can manage load turn-on and turn-off to prevent instantaneous large power transients, which can cause electromagnetic interference (EMI).

The load management center can contribute to reduced weight by reducing peak-power requirements by controlling power utilization, and can increase reliability by monitoring generator voltage and temperature.

The heart of PGS is the 270-volt dc generator (HVDC), which will ultimately replace both the 115-volt 400-cycle ac and 28-volt dc prime power sources in present aircraft. Replacing 28 volts dc with 270 volts dc reduces power-line current by almost a factor of 10, which in turn reduces power-line copper by a factor of 30 to 40. This weight factor is not totally reliable because of wire insulation, but a sizeable reduction is still evident. The 270-volt dc generator eliminates the problem, and weight of frequency control, in the present 115-volt 400-cycle generators. The 400-cycle ac generator is driven by a hydraulic

motor via a hydraulic pump and a power takeoff from the aircraft engine. The 270-volt dc motor can be driven through gearing directly from the aircraft engine, eliminating the hydraulic equipment, which adds weight and decreases reliability. The dc generator also eliminates the phase-lock problem of tying two ac generators together.

The bus contactor (BC) is also part of the PGS system and is used to switch power load above 10 amperes. One proposed design of the bus contactor is a hybrid approach (electromechanical in parallel with solid-state switches), which can resolve the problem of the arcing of mechanical contacts and the power dissipation in the solid-state device. The bus contactor must protect the related generator to provide shut-down in case of generator or load malfunction. Another proposed design of the bus contactor incorporates the Transcalent solid-state power device.

The blocks of the SOSTEL system consist of the multiplexers which interface with the solid-state transducers and other input devices, the demultiplexers which interface with the power controllers, and the SOSTEL processors. The blocks of the GPMS consist of the cable control units and the 1553 bus interfaces to the SOSTEL and other user subsystems. These modular blocks provide the standardized power modules, which can be used in all aircraft power systems.

### 9.3 Comments on the AAES Approach

#### 9.3.1 270 VDC

As presently defined the 270-volt dc generator is a basic building block of the AAES system. The 270-volt dc generator has already been added to MIL-STD-704C. This specification defines the maximum transient (over 50 microseconds) for a 270-volt dc system as 475 volts for 10 milliseconds. Filters or suppressors must be used to lower the transient voltage to levels acceptable to internal digital electronics requirements.

The high dc voltage poses potential problems with maintenance, insulation, and corona (see MIL-W-5088F) within densely packaged electronic modules, even though the Paschen minimum for corona at 270 volts dc is 100,000 feet.

#### 9.3.2 Power Generators

The AAES references have identified the samarium cobalt dc generator. and its power-to-weight characteristics as an AAES generator candidate.



The samarium cobalt generator is a prototype development, and even if it were developed, the limited world supply of samarium and cobalt could still be a limitation to its extensive use for power generation. This does not constitute a problem for assessing AAES as long as a weighting factor is applied when determining the weight-saving ability of the dc 270-volt generator. A fallback position is still the conventional ac generator with a diode bridge.

If the hydraulic power used to drive the power generators is shared by other systems using hydraulics, then the savings in going to dc may not be as great as they first appear.

#### 9.3.3 Semiconductor Devices

The 475-volt transient does constrain the semiconductor devices in the contactors and the power controllers to very high breakdown voltages. High-breakdown voltage (in the order of 500-volts and above) does limit the availability of transistors, and does increase transistor cost. The article "Power Controller Overview - Status and Trends," by Triolo, Marek, and Perkins, indicates some of the problems resulting from the high-voltage and high-current requirements. When all the requirements are implemented, the simple power controller can be fairly complex. MIL-P-81653A, "Power Controller, Solid State, and General Specification For", does impose high-voltage test(s) on power controllers (600 volts), which makes the selection of solid-state devices more difficult.

#### 9.3.4 DC to DC Converter

Most integrated circuit logic operates at fairly low voltage (5 volts, and may go to 4 volts in future devices). Pulse-width modulated dc-to-dc conversion from 270 to 5 volts, at very high current, may increase the generation of EMI. EMI can be decreased by using a transformer in the dc-to-dc converter design, but this may result in additional weight. The dc-to-dc converter can operate at fairly high frequencies, so that the transformers will not be as large as would be required in a 400-cycle power conversion.

#### 9.3.5 Contactors

The use of semiconductors in the contactors and the power controllers will reduce the EMI caused by arcing or bouncing in the relays presently used in aircraft. However, the EMI caused by current surges with turn-on and turn-off will still be present in the proposed AAES

design. Therefore, EMI suppression circuits and the related magnetics that are required by these circuits will still be required in AAES.

High-voltage relay contacts depend on some arcing to clean corrosion from the contacts. Low-voltage low-current relays are referred to as "dry" circuit relays, and are of special design to avoid corrosion. The hybrid contactors, with the semiconductors in parallel with the contacts, may possibly experience long-term contact problems.

With the metal contacts and semiconductors in parallel, it will be difficult to determine if one of the units has failed, resulting in a shortened life of the hybrid contactor.

#### 9.3.6 EMI and Lightning

The application of digital electronics in the GPMS and SOSTEL subsystems of the AAES makes the power subsystem of the aircraft much more susceptible to EMI, lightning, or other transient failure conditions than present aircraft power systems. The techniques developed for reducing the susceptibility of digital computers must also be imposed on the AAES.

#### 9.3.7 The MAP, AAES Interface

There are several aspects of the modular avionics packaging (MAP) program that do not presently seem to be integrated with the AAES concept. Figure 9-3 is taken from a MAP presentation. As presently conceived, the power cable enters from a single point at the top of the cabinet. With AAES, several power cables are envisioned, and (to account for battle damage) redundant power cables must be distributed and not occupy one area in the aircraft.

Figure 9-4 is an interior view of the MAP cabinet, which shows power supplies running down the center. Power controllers, EMI filters, GPMS multiplexers, GPMS demultiplexers, GPMS cable-control units, and some power-line contactors are required in the racks in order to provide power management. These modules must be located in the rack so as to have access to the power lines and to the AAES communication cables. This tends to restrict their placement. Redundant power and communication cables further complicate the physical module location problem, especially if damage tolerance is considered.



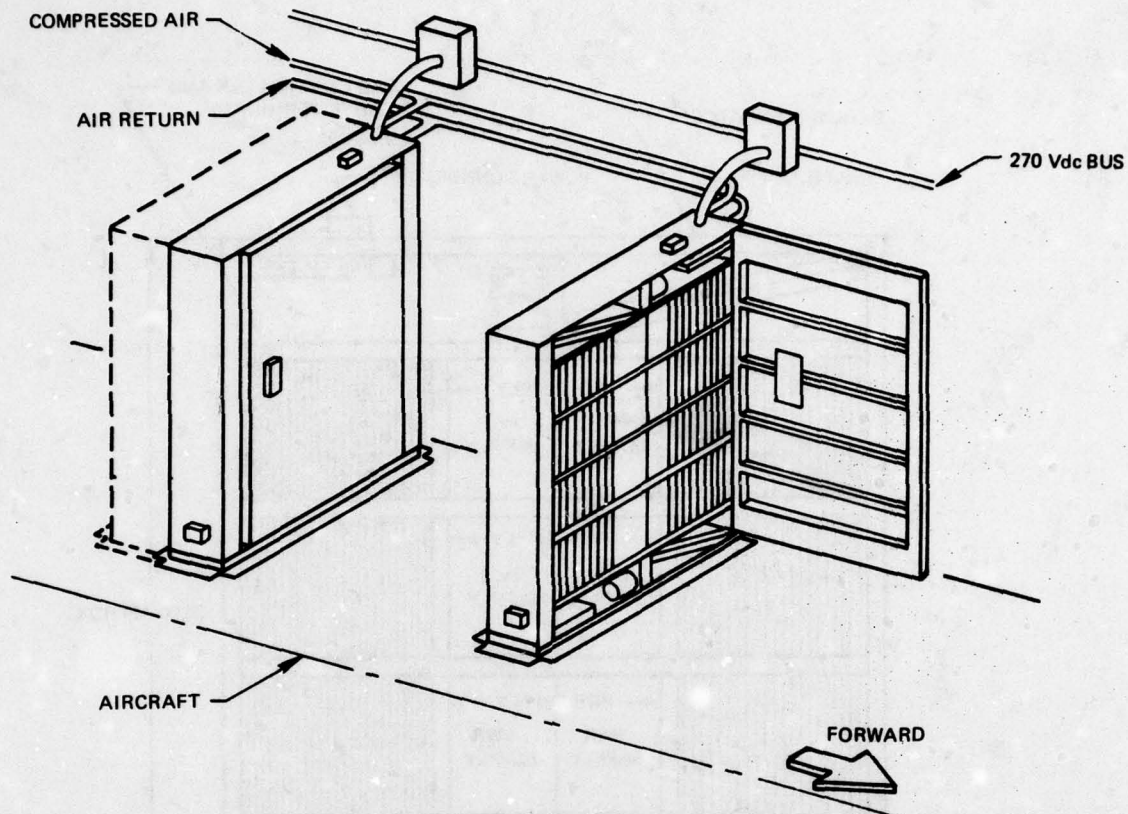
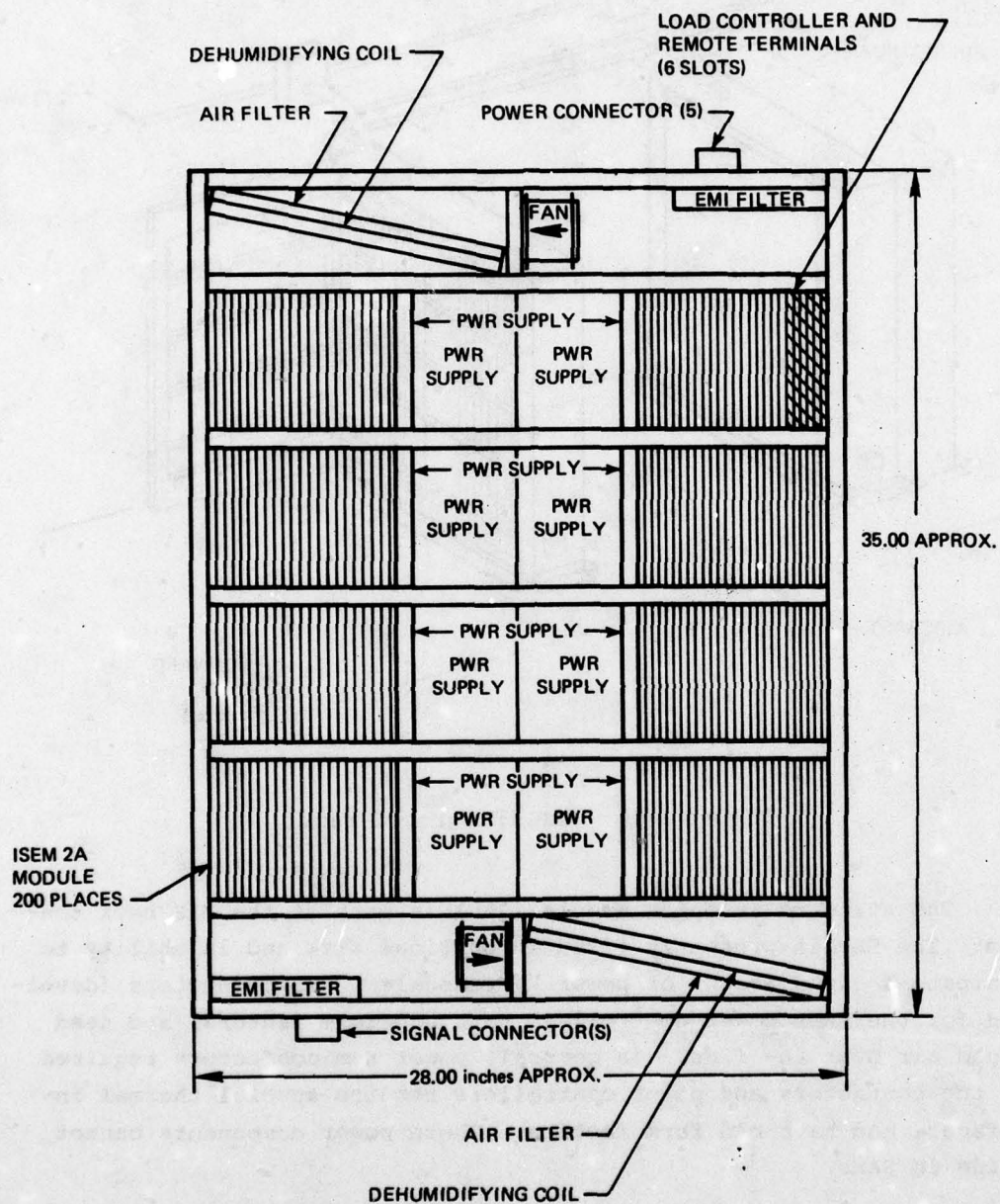


Figure 9-3. MAP integrated rack.

The standard avionics module (SAM) is part of the MAP rack concept. The SAM is presently fixed in physical size and in ability to dissipate a given amount of power. Transcendent semiconductors (developed for the AAES power controller) have odd form factors, and need forced air over the fins. In general, power semiconductors required for the contactors and power controllers require special thermal interfaces, and have odd form factors. These power components cannot reside in SAMs.

#### 9.3.8 Grounding and Shielding

The traditional approach to power distribution in aircraft systems allows the use of structure as the power return. This approach is in conformance with MIL-W-5088F and most other aircraft wiring



NOTE: DEPTH OF RACK INCLUDING ACCESS  
COVERS (NOT SHOWN) IS APPROX.  
5.00 inches.

Figure 9-4. MAP rack interior view.



standards. This method of power distribution has been satisfactory for present-day aircraft, but for future aircraft with flight-critical electronics it should be reevaluated.

Missile and spacecraft systems have imposed restrictions on power ground returns, shield grounding, and isolation of power returns from the chassis of subsystems. These restrictions were imposed to improve electromagnetic compatibility (EMC), to make the systems more tolerant to electromagnetic pulse (EMP), and in some cases to tolerate lightning.

The AAES should consider these techniques in the initial design, since denser packaging will increase the EMI problem. In addition, the use of composite materials in the structure will provide less shielding for lightning and other external interferences.

#### 9.3.9 Power for SOSTEL and GPMS

The problem of providing fault-tolerant power to these systems has not been addressed in the references.

#### 9.4 Summary

The AAES concept for load management using advanced technologies is consistent with a fault-tolerant avionics architecture, but some of the system integration requirements imposed upon the electrical system by flight-critical avionics needs more consideration. Examples of the integration requirements that should be considered when designing the power system include: EMC/lightning or EMP, potential for corona in high-density packaging, problem of power conversion and ground isolation in each functional subsystem, power distribution for redundant subsystems, and filtering required to provide continuous power to digital computing equipment.

If the AAES is to be used to power flight-critical electronics, it must be considered flight-critical also. In this case, the redundancy management and fault-tolerance of the FTMP is applicable to the AAES control system.

#### LIST OF REFERENCES

- 9-1 Vugraphs on the Advanced Aircraft Electrical System, Naval Air System Command, Washington, D.C.
- 9-2 "Overview of the Advanced Aircraft Electrical System (AAES) A-7E Prototype Design," by J.R. Perkins and A.J. Marek in NAECON, '77 Record.
- 9-3 "Power Controller Overview - Status and Trends," by J. Triolo, Naval Air Development Center and A.J. Marek and J.R. Perkins, Vought Corp. in NAECON, '76 Record.
- 9-4 "Electrical Load Management for Advanced Aircraft Electrical Systems," by D.E. Lautner and J.R. Perkins of Vought Corp. in NAECON, '77 Record.
- 9-5 "Transcendent Solid State Power Devices," S.W. Kessler, R.E. Reed, H. Shoemaker, K. Starter of RCA under Contract DAAK02-69-C-0609 and DAAK02-72-0642 for U.S. Army Military Equipment Research and Development Command (MERADCOM), Fort Belvoir, VA.
- 9-6 "Advanced Aircraft Electrical System (AAES) A-7E Prototype Design," Vought Corp. Final Report by J.R. Perkins et al under Contract N62268-75-C-0391 for Naval Air Development Center, Department of the Navy.



## SECTION 10

### PACKAGING

#### 10.1 Introduction

The packaging requirements for an integrated fault-tolerant avionics system have been reviewed. The packaging system must not degrade the capabilities provided by the fault-tolerant architecture. This implies the following special requirements:

- (1) Connectors and interconnections must not introduce the potential for single-point failures.
- (2) Environmental conditions such as temperature, vibration and battle damage must not introduce the potential for correlated failures, which can defeat the fault tolerance.
- (3) Redundant signal and power interconnects need to be testable.
- (4) Redundant power sources need to be integrated into the module and interconnect design.
- (5) The failure mechanism and failure history of all components of the packaging system must be known such that redundancy can be provided where necessary to prevent single-point or correlated failures. That is, the avionics packaging needs to be well integrated with the detailed implementation of the fault-tolerant architecture.

Avionics packaging often has received only token attention despite its undisputed impact on performance, cost, and weight. Protecting and interconnecting electronic components and providing for heat transfer are the primary packaging functions. These functions typically cost more, and typically consume several times more volume, than the electronic components in the system. The existence of the Navy sponsored Modular Avionics Packaging (MAP) program is evidence of concern in this important area, and of the possibility of a significant breakthrough in avionics design philosophy. The opportunity now exists for

a quantum step in avionics packaging. The size of this step will depend upon the wisdom and foresight exhibited in the final standards selection.

The MAP program is as yet young, and it is believed that some changes in emphasis will prove to be valuable. The special requirements of fault-tolerant avionics, power systems and data buses are of primary concern. Significant efforts are being expended on system architectures which will enhance reliability, maintainability, and minimize system life-cycle cost. Concurrent with the increased emphasis on fault tolerance, there is an opportunity for the MAP program to make timely advances in avionics packaging.

#### 10.2 MAP Program

An industry briefing on MAP was held at the Naval Avionics Center on 9 May 1978. It is the program as presented at that point in time which is commented on here. There are some substantial differences between the 9 May 1978 position and concepts which were embraced earlier in the program. For example, there was considerable effort expended in determining what standard module family could best be used in ATR boxes. This position has changed to a single-module size housed in an integrated rack. As another example, the standard module, ISEM-2A, was selected based on using dual-in-line packaged components. The present concept uses ceramic chip carriers, which increases the possible component density on the module by about a factor of three; however, the module size and pinouts have been kept the same.

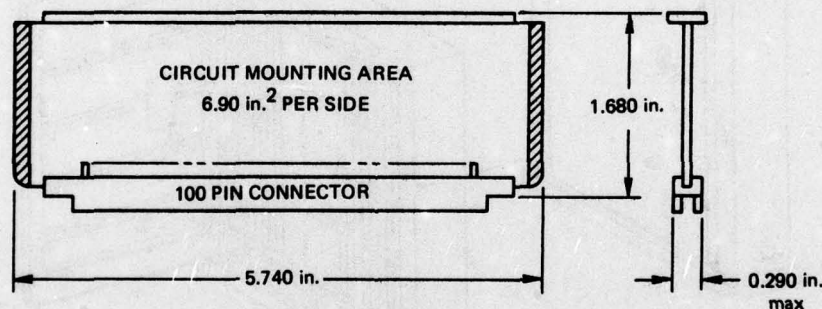
The MAP Program started with the concept that standardized modules shall be used for functional standardization. It is now also assumed that an integrated rack will be used. This rack will contain collections of modules without individual boxes enclosing functions, as has been past practice.

MAP objectives are to develop an avionics packaging approach which minimizes system life-cycle cost, significantly reduces system weight and size, maximizes system reliability, and enhances system maintainability. Also the module standardization program is to be established based on the multisystem use of common functions. An order of magnitude increase in reliability is to be the product of increased thermal efficiency and standardization. An order of magnitude increase in maintainability is to be brought about through the



use of built-in test (BIT) techniques. Avionics weight and volume are to be reduced by 30 to 50 percent, and supportability costs by 25 percent.

The current MAP program emphasis is on the standard avionics module (SAM), integrated rack concepts, and thermal management. The present choice for the SAM module is the ISEM-2A module shown in Figure 10-1. The connector uses the spade and tuning-fork concept, but work is going on to reduce the insertion force of these connectors. Emphasis will be placed on the use of hermetic leadless packages (chip carriers) for packaging silicon devices and hybrid circuits. These in turn will be mounted to a wiring board on the module. The module has been rated for 10 watts if conduction cooled, and 14 watts if air is allowed to impinge directly on it. Capability for both of these methods of cooling are built into the integrated rack. Studies are in progress comparing the use of vapor-cycle and air-cycle refrigeration, and fuel and ram-air heatsinks in various combinations for cooling the electronics in the integrated racks.



▨ CONDUCTION THERMAL INTERFACE AREA = 0.40 in<sup>2</sup>

Figure 10-1. Standard avionics module (ISEM 2A).

The integrated-rack concept does away with individual boxes surrounding functional groups of modules with the object of reducing weight. It consists of a cabinet where various primary backplanes are plugged into a secondary backplane, which distributes power and interconnects the primary backplanes. Modules are plugged into the

primary backplanes. The primary backplanes are analogous to the old box motherboards, and the secondary backplane takes over much of the old rack harnessing function. Figure 9-3, and 10-2 through 10-5 show various aspects of the integrated-rack concept.

It is interesting to note that the integrated-rack concept is much like the proposals made by the 1970 Airlines Electronic Engineering Committee for the New Installation Concept (NIC). This proposal has been shelved for now because airlines have experienced too much handling damage to unboxed modules and because it does not provide for an evolutionary transition from ATR box packaging.

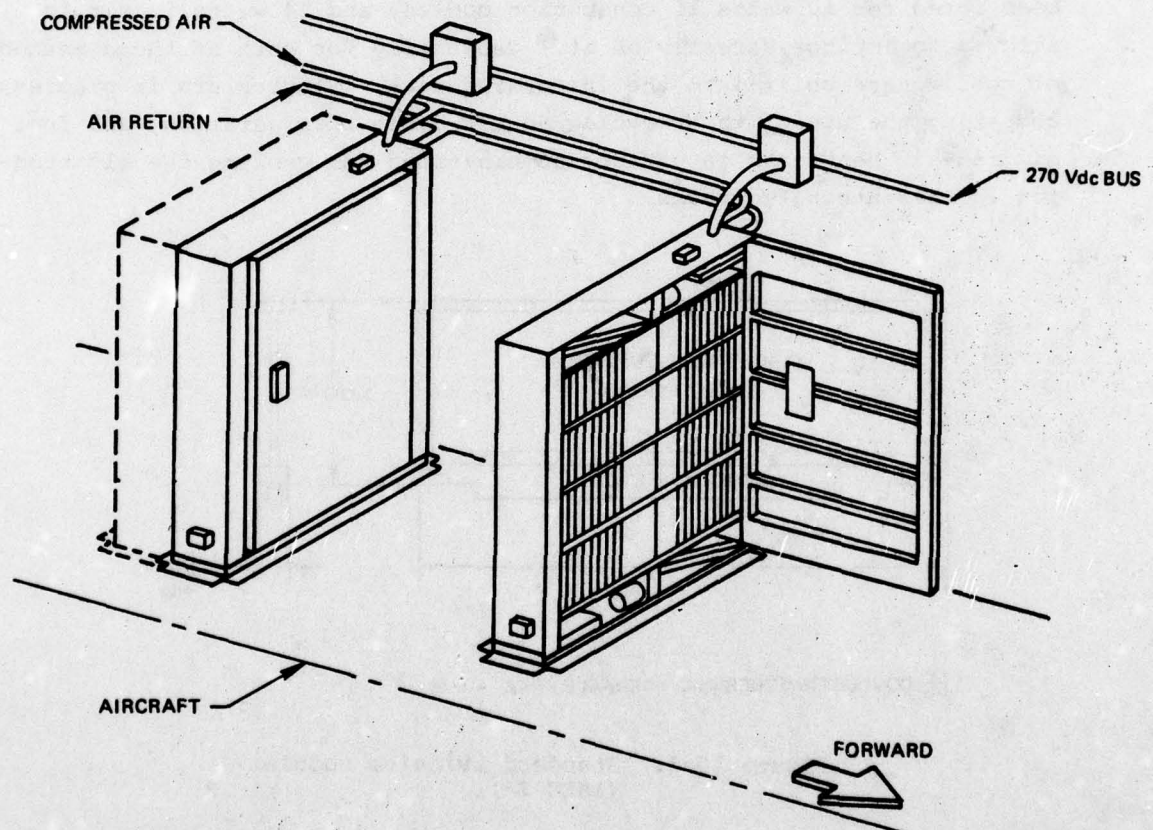


Figure 9-3 (Repeated). MAP integrated rack.



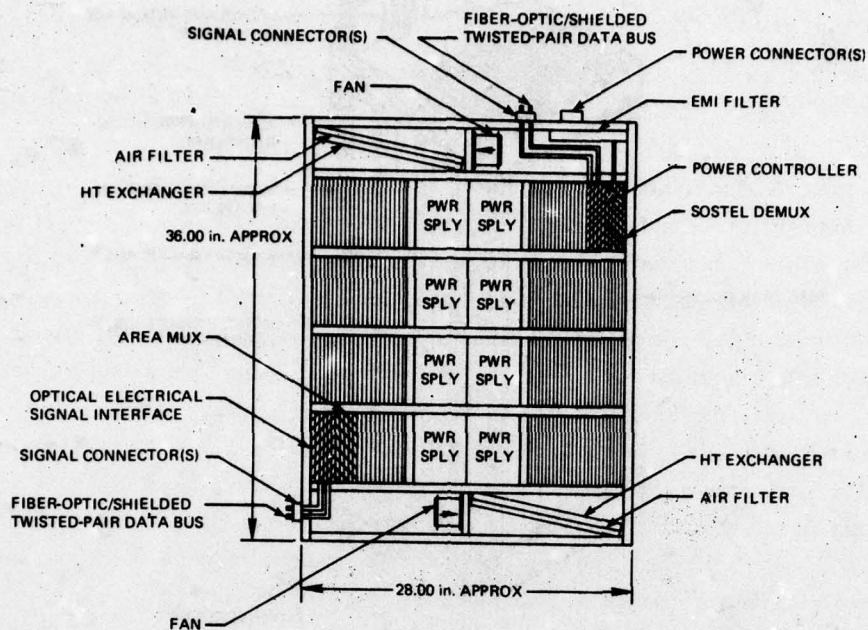


Figure 10-2. Integrated-rack concept.

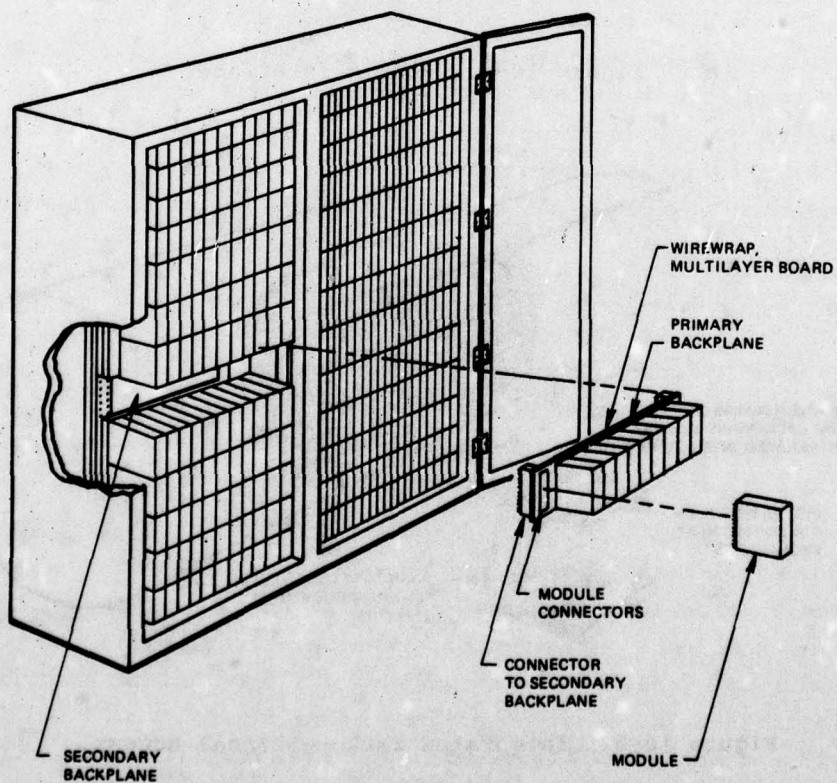


Figure 10-3. Backplane identification.

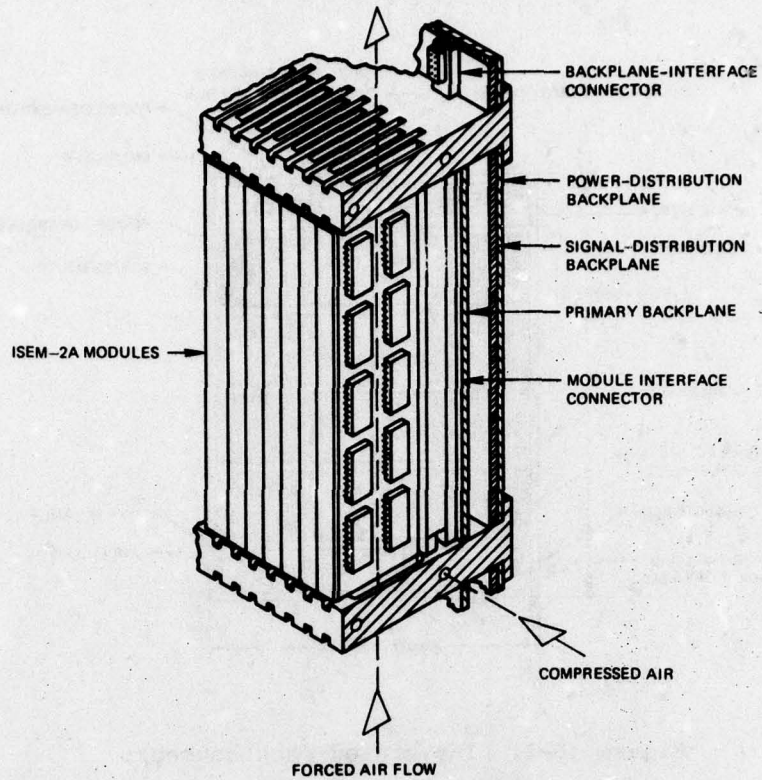


Figure 10-4. Module interface.

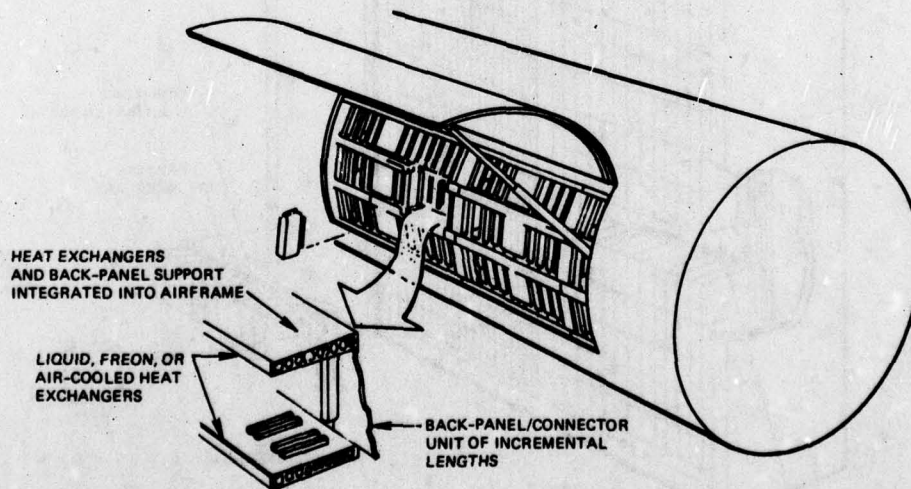


Figure 10-5. Integrated rack—external access.



### 10.3 Discussion

At the outset of a program such as MAP, the packaging problem and goals should be as clearly defined as possible. Where definition is not possible, the alternatives should be clearly stated. For example, one area which has not been defined is the possible need for protection of modules from salt spray. To determine this requirement, one must know the conditions under which the aircraft avionics will be repaired. Indeed, when making deck repairs, if modules are to be removed from outside the aircraft on deck in high winds, there is the possibility of contaminating the whole rack, as well as the removed modules. Module replacement from outside the aircraft would also preclude the inflight module replacement, which has been done in the past. Expanding on this theme, the total set of environments for the avionics should be defined including shipping, storage, and repair environments. Repair environments can include hangar, small air-capable ship, ground, etc. In addition, the move toward a distributed system will require that some avionics be placed in remote locations in the aircraft with vastly different environments than those encountered in a pressurized equipment bay. Also, there may be unusual form-factor limitations. These environments and form factors must be defined. the MAP program does not appear to address the distributed electronics concept or the effects of fault-tolerant architectures. It has been suggested in MAP presentations that throw-away modules are a desired repair method. However, there was no definition of the economic breakpoint where a module becomes too expensive to throw away, nor were there projections of MAP module costs.

In defining the ground rules/constraints one should strive for a minimum set. One of the ground rules given for MAP is that "the integrated rack will be the primary avionic enclosure." There was no evidence given of a study which demonstrated that an integrated rack is the best enclosure that can be used to satisfy the packaging requirements. Perhaps the use of the integrated rack should not be part of the ground rules/constraints.

Goals for the MAP program have been stated, but they have not been stated to emphasize the tradeoffs required. Packaging standardization is one of the tools necessary to optimize a set of avionics characteristics including size, weight, maintainability, reliability, cost, and the like. (Other tools include component standardization, system simplification, and repair philosophy.) Packaging standardization can provide lower cost, higher reliability, and easier maintainability; the price paid for these improvements is increased weight and

volume over that which could be achieved using the same packaging technologies with no standards. It is therefore suggested that this trade-off between standards and weight be emphasized and quantized in future MAP studies. Computer modeling will be required to optimize the complex tradeoffs involved.

Some of the MAP concepts will require a wider scope of study than that given to date. For example, the burden of increasing reliability has been put entirely on improved cooling. Certainly other environmental considerations, plus reliability improvements for components and improved assembly procedures, should also be addressed. The burden of improvement in maintainability has been placed on BIT. The overhead for this has been estimated at 5 to 15 percent of the electronics. However, the dependence of this overhead percentage on module size and, indeed, even the size of the overhead, has not been demonstrated. The stated goal of a 30 to 50 percent reduction in volume and weight over currently deployed avionics will likely occur from increasing large-scale integration (LSI) complexity, advances in circuit design, and other component improvements with no contribution from improved packaging. The question is whether, in the face of increasing electronics complexity, the need for redundancy, and the increased criticality of weight, this goal should not be more ambitious. It is stated that MAP must be compatible with 1985 components and beyond. There is evidence of component-development projections, but not much evidence of their impact on the packaging problem.

The present MAP position of a single-module size and a single-integrated-rack design concept capable of handling the highest power circuitry represents a position of maximum standardization. It also implies minimum flexibility, and maximum weight penalty. Because weight and volume penalties in avionics are far more critical than in land- or sea-based electronics, it is important to examine the possibilities for, and advantages of, less rigid standards (such as multiple module sizes), and other standards (such as standardizing the hybrid circuits). The selection of packaging standards must address the cases of high-frequency and low-frequency circuits, high power and low power circuits, logic and analog circuits, noise-sensitive, and not so noise-sensitive circuits, etc. The present standards emphasis appears to be limited to digital-circuit packaging. The MAP program should also address the anticipated environmental and form-factor



problems of distributed electronics such as embedded microprocessors, preamplifiers, and node electronics. These will be associated with sensors, effectors, antennas, and buses scattered throughout the aircraft, and it may be best not to confine them to a standard module size.

Component interconnections on the silicon chip are most efficient: hence, the push to larger and larger integrated circuits. Interconnections become progressively bigger, more expensive, and worse electrically as they progress through hybrid circuits, module boards, backplanes, and wiring harnesses. It is therefore suggested that more emphasis be placed on making connections at the hybrid-circuit level inside the chip carriers. In fact, a strong standardization program at the chip-carrier level, for hybrid circuits in chip carriers, will offer the same order of benefit as standardization at the module and rack levels. Integrated circuits are already at  $10^5$  gates/chip in the laboratory, and by 1985 chips with  $10^6$  gates will be available. With these tremendous strides in chip complexity, the benefits of commonality and standardization at the hybrid-circuit level should be fully exploited.

The ISEM-2A module, now proposed for the new SAM module, contains only one significant technology change over the old standard electronics module (SEM) modules. That is the low-insertion force connector. Yet this module standard, if adopted, should be used for at least the next 20 years. There are several concerns over this module choice.

Computer-packaging engineers are projecting the need for about a threefold increase in module connections over the next decade to accommodate a tenfold increase in module complexity. The use of redundant interconnection systems in future aircraft will further increase this need. If these projections have any validity, the 100-pin connector will not be adequate, and the retention of 100-mil connector-pin spacings will be costly in volume and weight.

Also, the long and low module form factor makes for inefficiencies in packing density at the rack level. It is interesting to note that if the connector pin density were doubled, the connector and module could be a little over half as long (considering guide pins) and still maintain the pin out. Then, the component area of the module could be maintained by doubling its height. The integrated-rack volume for interconnections and cooling will remain about the same. The end result

could be about a 30 percent increase in electronics-packing efficiency at the rack level.

It may be necessary to provide module protection beyond that provided by SEM modules because of tougher handling requirements, salt spray, and the possible need for barriers to failure propagation in fault-tolerant equipment. Failure propagation through fire may also require that fire-resistant materials and components be used.

Another area of concern is the single-size module. There are several SEM module sizes which evolved in programs where weight and volume were not nearly so important. The need for multiple sizes will be even more important in avionics to achieve high-density packaging, and because of a wider variety of components (like power semiconductors and RF components) and the problems of partitioning the many functions into modules.

The integrated-rack concept was developed primarily to save individual box weight and improve cooling. As presently envisioned, less than one fourth of the rack volume is devoted to active module volume. It would seem prudent to compare this with what could be achieved with a more innovative rack and box design. Another concern with the integrated rack is the means of reconfiguration. Subsystem collections of modules would have to be removed along with their backplanes, and other backplanes and modules installed. See Figures 10-4 and 10-5. The reliability of electronics handled in this manner can only suffer. Shielding and filtering between various functions, which are important to the electromagnetic compatibility, will be more difficult in the integrated rack than in the older box concept. The proposed methods for providing the necessary shielding should be tested at an early date. The degree of battle-damage protection that the more conventional approach provides will be reduced because a single event can impact more functions.

Other areas requiring more detailed work are the marriage of various backplane technologies and the interconnections between planes. Particular effort must be invested in the introduction of optical transmission lines in the backplanes, and also the interconnection of optical and matched impedance lines between backplanes and to modules. Of course, it may prove to be more prudent not to use optical transmission in the backplanes.



AD-A065 136

CHARLES STARK DRAPER LAB INC CAMBRIDGE MA

F/G 1/3

AN INTEGRATED FAULT-TOLERANT AVIONICS SYSTEM CONCEPT FOR ADVANC--ETC(U)

FEB 79

N00019-78-C-0572

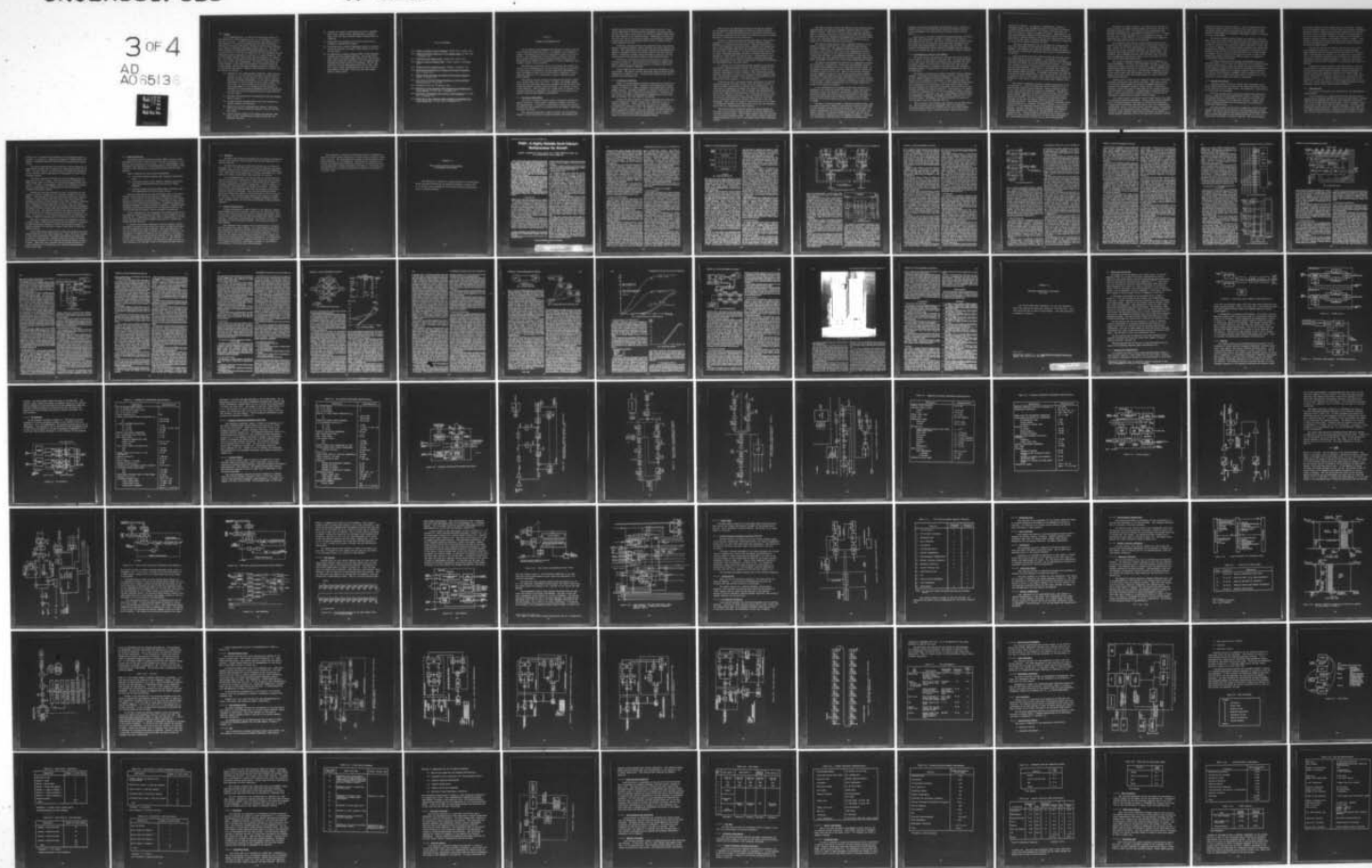
UNCLASSIFIED

R-1226

NL

3 OF 4

AD  
A065136



#### 10.4 Summary

It is recommended that, in light of the significant amount of data generated on the MAP program to date, it is worthwhile to back up and take a new look. In order to further define the problem, the new look should fill in missing information such as repair environment, effects of distributed electronics, effects of fault-tolerant architecture, effects of component development beyond 1985, etc. The new look should reassess the integrated rack and SAM compatibility with SEM. (It might be nice to have the two compatible, but the advantages and disadvantages should be quantized to find out if it really would be.) The new look should identify those factors that are important for comparison. For example, producibility might best be handled as part of life-cycle cost, rather than independently. The factor, module weight per circuit package, is not nearly so interesting as avionics weight per circuit package, because some module designs are much more difficult to interconnect than others. Specific tasks suggested to complete this new look are as follows:

- (1) Complete the set of MAP boundary conditions. More work must be done on: (a) the effects of projected component developments on module circuit density, and in turn on interconnection and thermal requirements; (b) The special packaging requirements of fault-tolerant avionics, power systems, and data buses; (c) factors affecting the system and module repair philosophy; (d) the packaging and environmental requirements for distributed and embedded electronic components; (e) costs and aircraft performance penalties associated with avionics weight.
- (2) Determine the desirability of standardizing leadless-packaged hybrid circuits.
- (3) Consider multiple standard module sizes, and reconsider increased connector-pin densities.
- (4) Demonstrate that the integrated-rack concept is more efficient than the best innovations applied to the old rack and box concept.
- (5) Perform detailed studies of the primary and secondary back-plane, wiring, interconnection, power distribution, and shielding functions.



- (6) Consider the impact of distributed electronics, embedded electronics, network nodes, redundant buses, redundant power lines, and fault-tolerant computer architectures on packaging.
- (7) Investigate what fraction of the total avionics can be packaged in candidate MAP formats.
- (8) Investigate how an order of magnitude increase in reliability can be achieved in avionics, which cannot be put into the MAP format.
- (9) Do quantitative tradeoff studies between various packaging concepts utilizing computer modeling. This study should start with an anticipated V/STOL avionics set, the environmental and operational constraints, and the packaging goals. Preconceived notions such as the integrated rack and compatibility with the SEM program should not be used. This will allow a clearer understanding of the cost of making SAM compatible with SEM. Select the final MAP concept based on these studies.

#### LIST OF REFERENCES

- 10-1 Report on Modular Avionic Packaging, TR2240, NAC, 3 August 1978.
- 10-2 Standard Electronic Modules, FY 1976 Summary Report TR 2146, NAC, 1 September 1976.
- 10-3 Standard Avionic Module Study, Vought Corp., March 1978.
- 10-4 Modular Avionics Packaging (MAP), General Electric 30 November 1977.
- 10-5 Modular Avionics Packaging Study, Lockheed, 9 November 1977.
- 10-6 Manufacturing Technology for Hermetic Chip Carrier Packaging, RCA, 23 August 1978.
- 10-7 Manufacturing Technology for Hermetic Chip Carrier Packaging, Hughes, 23 August 1978.
- 10-8 Heat Pipes to Improve Thermal Management of Avionics Systems, McDonald-Douglas, 9 July 1978.
- 10-9 NAVMATINST 4120.102C, 15 February 1978.
- 10-10 Evaluation of Low Insertion Force Connectors for Standard Electronic Modules, TR 2209, NAC, 12 December 1977.
- 10-11 Evaluation of Commercial Low Insertion Force Connectors, TR 2208, NAC, 12 December 1977.
- 10-12 Evaluation of Zero Insertion Force Connector for Standard Electronic Module XN-1, TR 2207, NAC, 12 December 1977.



## SECTION 11

### SUMMARY AND RECOMMENDATIONS

Key conclusions and recommendations developed in earlier sections of this report are summarized herein. The summary of conclusions focuses on the baseline core avionics system design, which is briefly described. The recommendations relate primarily to ways in which the Navy's ongoing avionics-related technology program efforts could be channeled to best support the development of a highly integrated, fault-tolerant, generic, avionics system.

The baseline architecture, which is configured with enough flexibility to be incorporated into any future Navy aircraft or retrofitted in CILOP fashion, is designed to provide substantial improvements in fault and damage tolerance and in maintainability compared to current generation avionics. It is based upon a totally integrated digital design approach.

The array of vehicles to which the baseline core system is applicable includes CTOL, VTOL, and V/STOL. Of these, the demands imposed on avionics by V/STOL-type vehicles are probably the most stringent with V/STOL requirements tending to pace the design. As a result, the particular needs of V/STOL are dealt with in some detail in certain of the following paragraphs to demonstrate the adequacy of the baseline core system to meet the most demanding requirements.

#### 11.1 Functional Requirements

The core of a generic avionics system is defined to include four primary functions: flight control, navigation, display and control, and communications. In addition to these basic functions the core system must have the flexibility to interface with and support a wide array of mission functions.

The increasing performance, safety of flight, and survivability demands of future vehicles will require automation and extremely high

reliability and survivability of the flight-control function. Path-control modes such as autoland, terrain avoidance, and various weapon delivery tasks require high levels of interaction between flight control and other elements of the avionics system. Integration of flight control with the rest of the avionics system is an inevitable consequence of both future mission requirements and the need to minimize overall system complexity and cost.

Navigation requires gathering and integrating data from numerous sensors to generate accurate, timely estimates of aircraft position and velocity. The organization of navigation resources and redundancy management of navigation sensors are primary tasks of the core avionics system. Flexibility to allow growth and change is essential for easy incorporation of new navigation sensors as they become available.

Both flight- and mission-related functions must be supported by displays and controls. Multifunction use of displays to support both types of functions can provide high levels of reliability and survivability without excessive complexity.

Future communications systems will place heavy dependence on the JTIDS system. The generic avionics system must support mission-specific communication elements as well. Various radio-navigation aids such as GPS must also be provided for.

#### 11.2 Information Processing

Sharply increased demands on digital processing will characterize avionics systems of the 1990s. Dispersal of the computational system elements is necessary to reduce throughput requirements and to provide damage tolerance. Increased automation of critical functions will spur the need for very high reliability and survivability. Flexibility to allow growth and change, and appropriate standardization of both hardware and software, are each essential attributes.

Current and future advances in digital technology will yield significant reductions in weight, volume, power, and cost per unit of computing capability. These reductions will permit the dispersal of computation elements by allowing for embedded local processors in sensors, actuators, displays, etc. The resulting increased autonomy of these elements will produce significant reductions in the internal avionics data-communications bandwidth requirements, and will impose a natural partitioning on software development efforts.



While much of the computation system can and must be dispersal, the all-important system management or executive function should be centralized. A hierarchical organization of resources is required with well defined priorities and the ability to organize resources to respond dynamically to changing tactical situations.

A review of current avionics architectures shows a trend toward segregation of functions with multiply redundant hardware elements dedicated on a function-by-function basis. Numerous demonstrations of this n-tuple redundant, segregated-function approach have already been carried through flight tests, and a few operational aircraft incorporate this type of architecture. The Navy's Digital Flight Control System (DFCS) technology development program is currently pursuing this approach in a joint effort with the Air Force. Extrapolation of this architecture consisting of multiply redundant segregated subsystems, to the requirements of the 1990s, projects sharp increases in complexity and unacceptably large life-cycle costs.

In contrast to this segregated-function approach, an integrated architecture, embodying pooled resources that can be flexibly and dynamically allocated on a priority basis, is far better suited to the stringent requirements of future avionics. This architecture incorporates the high level of integration necessary to reduce complexity and cost, while embodying extensive fault detection and identification procedures which allow automatic system reconfiguration.

Although the highly integrated avionics architecture can achieve significant savings in weight, volume, power and life-cycle costs, it imposes stringent requirements on internal data communications. The very nature of integration eliminates the concept of segregated functions and dedicated subsystems, and the internal data-communications system must, therefore, be sufficiently reliable and survivable to entrust it with flight-critical data. Also, an integrated avionics system requires interconnection of numerous small units, as contrasted with a conventional system that typically interconnects a few large subsystems. Hence, both the number of interconnections and the volume of information transmitted are significantly larger for the integrated system than for a segregated-function approach.

The fault-tolerant information-processing system proposed for the generic avionics system baseline employs the pooled, dynamically allocated resource concept. Three primary elements constitute the baseline information-processing system: fault-tolerant computer complexes, embedded processors, and a fault- and damage-tolerant data-transmission network.

The baseline system contains two identical fault-tolerant computer complexes which are physically separated from each other to achieve damage tolerance. These fault-tolerant computer complexes represent pools of processors organized as flexibly configurable multiprocessors. They are designed for extremely high dependability of continuous computation, with transparent means of detecting and identifying faulty modules and of self-reconfiguration and recovery.

The baseline design employs local or embedded processors in the sensors, actuators, displays, etc., with which the fault-tolerant processors communicate. The embedded processors perform all local functions, such as built-in test and on-line compensation, and they provide the interface with the internal data-communications network. A significant advantage of the embedded processors is that they serve to distribute the total computation load and significantly reduce bandwidth requirements imposed on the internal data-communications system.

The familiar multiplexed bus structures are inadequate to provide the reliability, survivability, connectivity and throughput required for internal data communications of an integrated avionics system. In the baseline avionics system, internal communications are carried over a network of point-to-point links. Nodes of this network interface with the embedded processors in the various avionics system elements. Switches within the nodes route data transmissions over the links of the network. Node switching is under the control of the fault-tolerant processors which configure the switches to establish a virtual bus within the network, allowing communication with all system elements connected to the network as required.

A rich interconnection of elements allows great flexibility in configuring the virtual bus within the network. A very high level of tolerance to faults and damage is afforded by providing the fault-tolerant processors with the ability to quickly grow or reestablish a new virtual bus around failed or damaged elements, so as to reach survivors and continue supplying critical functions after a failure or damage.

The Navy's technology program in Information Handling System (IHS) is currently pursuing a number of areas directly relevant to this type of fault-tolerant avionics architecture baseline. The IHS effort, to develop a methodology for partitioning the avionics, can yield a systematic approach to delegating responsibility both within the fault-tolerant computer complexes and in the embedded processors. Of particular significance is the IHS effort to standardize microprocessor



languages, and software development and evaluation tools. Strong coordination should be established between these IHS activities and any future development of an integrated fault-tolerant generic avionics system.

Ongoing work in the Navy's AVIOPTICS technology program (to develop fiber-optic data-transmission systems) is relevant to the baseline data-communications network. The effort to develop point-to-point fiber-optic links can provide important design knowledge and experience for the needed further development of the network links and nodes. The avioptics high-bandwidth digital data-bus development may have application as the internal data bus for the fault-tolerant computer complexes.

### 11.3 Instrumentation, Control, and Guidance

The flight-control and navigation functions require data from inertial sensors, air data sensors, and radio aids. By taking cognizance of the total set of functional requirements to be satisfied, the types of elements likely to be available in the 1990s time frame, and the potential effectiveness of systematic fault-detection and redundancy-management methods, a configuration of minimal redundancy and cost can be established.

The baseline configuration includes a strapdown inertial reference assembly, incorporating four skewed navigation-grade ring-laser gyros and four skewed navigation-grade accelerometers. Although the primary function of this assembly is navigation, it serves pilot-display and flight-control functions as well. A set of four additional skewed gyros and accelerometers, of pilot-display and flight-control grade, serve the pilot-display and flight-control functions, but do not have the performance necessary for inertial navigation. This second set of sensors is physically separated from the first for purposes of damage tolerance. The total array of eight gyros and eight accelerometers together possess the performance, redundancy, and damage tolerance required for all the functions served. Furthermore, the graded redundancy and quality of the instruments yield low complexity and cost.

Radio-navigation aids include a GPS receiver and two JTIDS receivers. The navigation function implemented within the information-processing system combines the information from the inertial sensors and radio-navigation aids to provide accurate, timely estimates of

position and velocity. In addition, a comprehensive redundancy-management system (also implemented within the information processors) provides real-time comparisons of the outputs of various radio and inertial elements to detect and identify failures. Comparisons of outputs from dissimilar elements provide additional sources of information with which to identify failures. Thus, use of so called analytic redundancy allows reductions in hardware replication, resulting in reductions in total avionics system complexity and cost.

The Navy's IISA program is key to providing the technology necessary for development of the baseline strapdown inertial-reference assembly. The proposed IISA laser-gyro navigator flight-test program will provide an operational experience base in areas of performance and reliability.

Since the baseline avionics system utilizes the individual gyros and accelerometers as modular elements, it is assumed in the design that these instruments are configured as self-contained units possessing individual interfaces with the data-communications network. This assumption is based on very preliminary life-cycle-cost analyses for laser gyros and accelerometers. Furthermore, it is assumed that display and flight-control-grade instruments will be significantly less expensive than inertial-grade instruments. Procurement, maintenance and logistics costs must be better defined before these assumptions are justified. It is important that the IISA program promote competition between potential suppliers of laser gyros in order to best answer these important questions.

The concept of embedded processors (already being pursued by the IISA program) is an area of investigation which is essential to the fault-tolerant modular avionics approach. Another important question is the likely failure modes for laser-gyro instruments. Appropriate failure coverage and self-test methods require knowledge of how likely faults manifest themselves, and the IISA program could appropriately address this problem. Also to be resolved is the question of damage tolerance and the need for physical separation between inertial-instrument assemblies. The possibility of separating collocated rigidly coupled assemblies by a bulkhead, for damage tolerance, is a possible alternative to the physically separated configuration proposed as the baseline. Similarly important is further development of redundancy-management methods for comparison of inertial sensors with radio-navigation aids. Closely related is the possible use of radio aids to perform real-time in-flight calibration of the inertial instruments. The IISA program could be the focus for all these investigations.



In addition to inertial sensors, the flight-control function requires air data, and V/STOL aircraft depend critically on air data at low speed and during hover. The baseline system incorporates dual-redundant multipurpose air data probes for high-speed (i.e., > 60 knots) air data, and a triply redundant set of air-temperature sensors. For application to a V/STOL vehicle, a triply redundant set of omnidirectional low-speed sensors would also be required.

The integrated fault-tolerant avionics baseline does not have separate air data computers, as are found in conventional avionics systems. Rather, the air data computation task is performed, as one of many jobs, by the fault-tolerant computers. Air data pressure transducers for each probe interface with individual network nodes. Each transducer has an embedded processor which performs local functions, such as compensation and built-in tests, and also provides the interface with the associated network node.

The concept of embedded processors in air data transducers requires development. The Navy's DFCS development program is a logical focus for an effort to provide this technology. The DFCS program could appropriately address the areas of sensing, transducers, embedded processors, compensation, computation, and redundancy management—as they are relevant to both high- and low-speed air data. Significant development is required in the latter speed regime for future controlled-configuration vehicles.

Another area of concern is the interfaces between the information-processing system and flight-control actuators. Here again the baseline avionics system assumes the presence of embedded processors in actuators for aerodynamic surfaces, engine controls, thrust-vector controls, control jets, and any other types of force producing mechanisms with which it interfaces. An appropriately coordinated DFCS program in these areas could be of enormous benefit to the evolution of the generic integrated fault-tolerant avionics system.

For V/STOL aircraft, the entire area of flight control is in need of significant technical development. Flight control during transition, hover, and landing aboard a small ship are particularly significant. A broad range of mechanisms including control jets, lift augmentation, thrust-vector control, fan-pitch control, and many others, are possible candidates as force-producing control effectors. Similarly, questions of manual versus automatic control during transition and landing in adverse weather conditions, with all the associated tradeoffs involving

displays and controls, must be resolved. The firm technological base for making rational decisions in these critical areas does not exist. Significant efforts should be undertaken by the DFCS program, and/or the Navy V/STOL Capability Development (NAVSTOLAND) Program, to establish this base of knowledge.

The overall problem of V/STOL landing (and especially landing on small ships) is an even broader technology area involving path guidance, as well as flight control. Preliminary studies have indicated that very significant savings in aircraft gross weight are possible if hover time on landing is reduced. An automatic-landing procedure that accomplishes transition and touchdown in one continuous maneuver, eliminating a prolonged hover period, can be of enormous benefit to the V/STOL program. For example, such an innovation may allow an aircraft of about 35,000 pounds gross weight to perform the V/STOL A mission, as compared to 55,000 pounds currently projected.

A closely coordinated effort between the NAVSTOLAND program and the DFCS program, should examine the feasibility of a minimum-hover-time automatic-landing system. Among the issues that should be addressed is the question of arresting the aircraft at touchdown on the deck, and the requirements imposed by an arresting system in terms of matching aircraft and deck attitude at touchdown. Also the question of using the elements of the shipboard autoland system in portable form, to aid landings in forward combat areas should be addressed.

#### 11.4 Displays and Controls

A modular display and control concept under development by the Navy's AIDS program forms the basis for the display and control aspects of the baseline avionics system design. Two approaches to incorporating the AIDS system are considered.

The first approach simply attaches the AIDS system, essentially intact, to nodes of the data-communications network. Although this is the simplest method of incorporation, the result is not suitable for flight-critical displays of an operational aircraft. However, this Level-1 integration method could be used as part of an exploratory flight-test demonstration.

The other approach, called Level-2 integration, extends the fault-tolerant computer and network concepts much more deeply into the AIDS system. The current AIDS configuration, employing standardized programmable displays and keyboards with embedded processors is totally consistent with this Level-2 integration approach. The various display



and control elements are each assigned nodes in the network. Management of the display system is the task of the fault-tolerant information-processing system. Flight-critical display computations are also performed in the fault-tolerant computers, and data are transmitted over the fault-tolerant network. Displays and controls can be flexibly assigned to priority tasks in the presence of faults or damage. Thus, the fault-tolerant avionics architecture can bring high levels of reliability and damage tolerance to the display and control elements of the avionics system baseline.

Displays and controls for mission functions may require additional processing facilities, in the form of regional processors. These processors could be incorporated into the network in close proximity to the displays and controls. It is difficult at the present time to make definitive judgments about the need for or capability requirements of mission display processors. In any case, the network architecture has sufficient flexibility to incorporate them readily if required.

Although the current progress on the AIDS program is quite compatible with the integrated fault-tolerant avionics system baseline, some additional effort should be devoted to the AIDS interfaces. A program to define the embedded processors at the network nodes, for all the AIDS displays, keyboards, controls, etc., would provide vital design information. An important aspect of this task should be the designation of local and central information-processing tasks so as to define the display requirements for local embedded processors, regional processors, and the fault-tolerant computer complexes.

#### 11.5 Communications

Two levels of integration were identified for the communications system elements.

Level-1 integration of communications closely parallels the so-called black-box approach employed in current avionics systems. Separate communications subsystems are included and interfaced at individual nodes of the information-processing network. Thus, for example, nodes are assigned to GPS, JTIDS, UHF, and VHF subsystems. The information-processing system provides the integration of outputs from these various communications subsystems. Although the Level-1 system configuration is quite similar to current configurations, it differs in that a

higher level of automation and management of the subsystem outputs is provided. In particular, redundancy management of communications subsystems is automated within the fault-tolerant information-processing system.

The Level-1 approach does not provide the overall integration of communication elements that is desired for operational aircraft of the 1990s. It could, however, be used as part of a flight-test demonstration to provide a base of experience for development of a more highly integrated system.

The Level-2 approach integrates the various communications elements as well as their outputs. The Navy's Tactical Information Exchange System (TIES) is the basis for the Level-2 communications architecture. In effect, the management, control, and information processing of the TIES system are incorporated into the fault-tolerant information-processing complex.

Whereas the functional elements of Level-1 integration are subsystems, in Level-2 integration, the functional elements are far more modular. TIES consists of three major functional sections: frequency conversion, signal distribution and control, and signal conversion. Frequency conversion includes antennas, front ends, frequency converters, and intermediate frequency amplifiers for the  $L_x$ , UHF, VHF, and HF bands. Redundancy in each band can be tailored to mission requirements. The signal-conversion section comprises wide-band and narrow-band signal conversion units. The signal-distribution and control section manages the resources of both frequency conversion and signal conversion sections. In particular, it can route signals and data to reconfigure the system to respond to changing tactical situations, faults and damage.

The Level-2 approach has the potential for considerable savings in weight, power, volume and life-cycle costs over Level-1. Furthermore, the TIES architecture employed in Level-2 can provide the basis for flexible allocation of resources to support high levels of fault and damage tolerance.

Future TIES efforts should strive to more fully exploit the fault-tolerant potential of the TIES architecture. Also, GPS is not included in the current preliminary TIES concept and should be considered for inclusion in later versions. Another concept that might be reviewed is the current placement of the frequency-conversion elements at antenna sites. Additional fault tolerance may be afforded by pooling frequency converters and IF amplifiers to be flexibly switched between various front ends.



### 11.6 Power Distribution

The Advanced Aircraft Electrical System (AAES), currently under development by the Navy, forms the basis for power distribution in the baseline fault-tolerant avionics system. Active load management is the fundamental AAES innovation. Sizable benefits in reduced weight, higher reliability, and reduced cost are potential benefits from the AAES approach.

AAES is comprised of three primary developments:

- (1) A new Power Generation System (PGS) supplying high-voltage power.
- (2) Solid-State Electric Logic (SOSTEL), embodying solid-state load control switches which are activated by a SOSTEL processor.
- (3) A General-Purpose Multiplex System (GPMS), providing redundant communications for active power control.

While the AAES concept can serve the primary needs of the fault-tolerant avionics system, certain aspects of AAES must be examined further. In particular, the detailed design of AAES must take careful account of reliability and survivability. Single-point failure vulnerabilities must be identified and eliminated. A systematic approach to redundancy management of power resources is essential. Redundant power controllers and multiplex terminals will be required for many of the fault-tolerant-system elements, such as processors and memories.

Future development of the integrated fault-tolerant avionics system should be closely coordinated with AAES to ensure compatibility between the two. A potentially fruitful area of investigation is the use of the fault-tolerant avionics computers and network to perform the SOSTEL and GPMS functions.

A problem of particular importance to digital systems is power transients, and design of the active load control in AAES should emphasize the suppression of power transients. Other considerations are the use of high-voltage power in terms of its impact on packaging, the availability of materials (samarium) for the generators, the general availability of high-voltage solid-state devices, and the problems of dc-to-dc conversion for low-voltage digital devices. Also important are the use of shielded grounding, the isolation of power from the aircraft structure, and tolerance to EMI of the SOSTEL and GPMS functions.

### 11.7 Packaging

The Navy's MAP program was reviewed for the purpose of determining how the fault-tolerant avionics system could be made compatible with modular packaging.

The MAP program employs standardized modules and an integrated equipment rack. A prime objective is an order of magnitude increase in avionics reliability from increased thermal efficiency and standardization. Other goals of the program are to enhance maintainability, reduce weight and volume, and minimize life-cycle-costs of avionics.

It is recommended that a new look be taken at the MAP program in view of the needs and requirements of highly integrated fault-tolerant avionics architecture. Of particular significance are the impacts of projected component developments on packaging, in terms of module circuit density, interconnection, and thermal requirements. The possibility of multiple standard-package sizes and increased connector-pin densities should be considered. Also, the integrated-rack concept should be reviewed and compared to new innovations of the old rack and box concept. Fault-tolerant system aspects, that should be included in this review, are the requirements of embedded processors, network nodes, redundant power, and the overall architecture of fault-tolerant computers.

### 11.8 General Considerations

Mission requirements for Navy aircraft of the 1990s will impose severe demands on avionics systems. High levels of automation of both mission- and flight-critical functions will require much greater levels of reliability and survivability than are achieved with current configurations. Availability, maintainability, and logistics (as well as procurement) will be significant factors in the life-cycle costs of these systems.

The integrated fault-tolerant avionics architecture is designed from the outset to be highly modular. Its pooled resources and extensive onboard fault-detection and identification facilities provide required reliability and survivability with minimum complexity. At the same time, the onboard fault-diagnosis capability greatly reduces the arduous fault-tracing aspects of avionics maintenance. By specifically tailoring the maintenance and logistics procedures to the integrated fault-tolerant architecture, considerable increases in availability and decreases in life-cycle costs can be achieved.



Major advances in electronics technology promise a formidable expansion of future avionics capability for given levels of weight, volume, power, and cost. The integrated fault-tolerant avionics approach takes maximum advantage of this technology. The modularity, pooling of resources and extensive fault-detection capabilities of this architecture can simultaneously increase performance, availability, reliability, and survivability while minimizing maintenance, logistics, and ultimately life-cycle costs.

APPENDIX 3-A

FTMP—A HIGHLY RELIABLE FAULT-TOLERANT  
MULTIPROCESSOR FOR AIRCRAFT

This appendix is a reprint of a paper published in the Proceedings of the Institute of Electrical and Electronic Engineers, October 1978. It is the most concise description of the fault-tolerant multiprocessor that is currently available.



# FTMP—A Highly Reliable Fault-Tolerant Multiprocessor for Aircraft

ALBERT L. HOPKINS, JR., SENIOR MEMBER, IEEE, T. BASIL SMITH, III, MEMBER, IEEE,  
AND JAYNARAYAN H. LALA, MEMBER, IEEE

**Abstract**—FTMP is a digital computer architecture which has evolved over a ten-year period in connection with several life-critical aerospace applications. Most recently it has been proposed as a fault-tolerant central computer for civil transport aircraft applications. A working emulation has been operating for some time, and the first engineering prototype is scheduled to be completed in late 1979.

FTMP is designed to have a failure rate due to random causes of the order of  $10^{-10}$  failures per hour, on ten-hour flights where no airborne maintenance is available. The preferred maintenance interval is of the order of hundreds of flight hours, and the probability that maintenance will be required earlier than the preferred interval is desired to be at most a few percent.

The design is based on independent processor-cache memory modules and common memory modules which communicate via redundant serial buses. All information processing and transmission is conducted in triplicate so that local voters in each module can correct errors. Modules can be retired and/or reassigned in any configuration. Reconfiguration is carried out routinely from second to second to search for latent faults in the voting and reconfiguration elements. Job assignments are all made on a floating basis, so that any processor triad is eligible to execute any job step. The core software in the FTMP will handle all fault detection, diagnosis, and recovery in such a way that applications programs do not need to be involved.

Failure-rate models and numerical results are described for both permanent and intermittent faults. A dispatch probability model is also presented. Experience with an experimental emulation is described.

## I. INTRODUCTION

THE FTMP (Fault-Tolerant Multiprocessor) is a computer architecture that has been studied, simulated, modeled, and emulated extensively over the past several years. It is scheduled to be implemented in an engineering prototype form within two years of this writing. The principal goal of FTMP is to be extraordinarily survivable without being difficult to program, operate, or maintain. It is presently predicted that the overall FTMP failure rate will be less than  $10^{-9}$  failures per hour, provided that maintenance is available within no more than ten hours of dispatch. In most cases, however, it will not be necessary to maintain the FTMP at intervals of less than 200–300 hours.

The FTMP structure can be described as an arbitrary number of processor modules with local, or *cache*, memories, and an arbitrary number of memory modules, interconnected by redundant serial buses. Modules are associated into groups of three to perform triply redundant functions. All data is distributed synchronously and in triplicate, and every module contains a voting element to mask bus disagreements. All

modules contain special circuits to create logical and physical boundaries to halt the propagation of faults from one module to another.

The FTMP is intended for use as one of at least two central computers in a redundant distributed digital system designed to serve as a highly survivable avionics system [1].

### A. Background and Context

The development history of the FTMP dates to 1965, with a serial-bus multiprocessor concept for spaceborne control applications [2], [3]. Increasingly redundant versions were conceived, including one in 1969 intended to serve as a preliminary design baseline for a manned spacecraft, i.e., the space shuttle [4]. At that time, a concept was stated for the systematic design of a redundant, fault-tolerant vehicle, employing fault-tolerant "regional" computers, each of which was to be the master of an I/O bus connected to a number of dedicated (micro-) computers, local to each of a number of sensor and effector components or subsystems [5]. In the early 1970's, some of the basic concepts were tested by simulation in a laboratory multiprocessor arrangement called Cerberus. The National Science Foundation sponsored most of this testing effort.

There were two particularly significant outcomes of this work. One was a network I/O data communication structure to replace the topologically leaner, and therefore more vulnerable, I/O bus [6]. The second was a significant improvement in the redundancy management capability of the architecture [7], [8]. As a result of these developments, the Draper Laboratory undertook the construction of breadboard emulations of the new multiprocessor and the network as independent Research and Development projects. Evaluations of various aspects of these emulations were sponsored by the National Science Foundation, the Office of Naval Research, the NASA Langley Research Center, and Draper itself.

The Draper study concerned itself with the design of a robust integrated avionics systems concept suitable for control-configured aircraft, and numerous other life-critical applications. This concept was to use a fault-tolerant central computer with a second remote identical computer available to take over in case of damage to the first. The concept also used the I/O network as a fault-tolerant and damage-tolerant medium for maintaining access to all surviving system elements. The third prong of the concept was a redundant sensor and effector architecture, with algorithms executed centrally to determine which, if any, of the sensors and effectors were malfunctioning [9]. The entire system concept came to be called OSIRIS, (onboard, survivable, integrated, redundant information system, [10]).

Manuscript received March 1, 1978; revised May 12, 1978. This work was supported by the NASA Langley Research Center under Contract NAS1-13782, and the National Science Foundation under Grant DCR74-24116.

The authors are with the Charles Stark Draper Laboratory, Inc., Cambridge, MA 02139.

Meanwhile, NASA Langley sponsorship further developed the fault-tolerant multiprocessor architecture in the direction of civil transport aircraft application, along with a competing architecture developed at SRI International, called SIFT [11]. In 1977, a design specification was drawn up for an engineering prototype of the multiprocessor, to be built by a major avionics manufacturer. At this point, the name FTMP was adopted to signify this particular architecture and its derivatives.

The FTMP represents a major architectural advance beyond the contemporary practices of computer redundancy in aircraft systems. All too often, computers have been interconnected in the simplest possible way, leaving as a programming task the detection and isolation of each fault and the subsequent recovery. This approach has serious problems, including the means of granting authority to a valid module without granting it to an invalid one. It is also virtually impossible in such approaches to separate the redundancy management software from the applications programs, with the result that both are greatly complicated. Validation is a difficult problem in these systems.

The FTMP is quite different from some other fault-tolerant computers for different applications. A fault-tolerant spacecraft computer, for example, has a similar task, but a dissimilar survival requirement. Other fault-tolerant architectures are meant to serve general data processing tasks in a benign environment with maintenance available. The next subsection attempts to show how the architecture of the FTMP corresponds to the class of applications it is designed to serve.

### B. Rationale of the FTMP Approach

The intended use of the FTMP is to support critical control functions in vehicles, process plants, life-support, or any similar application in which maintenance is available periodically or after a delay, and where loss of control leads with significant probability to high cost in terms of life or property. The failure rate at the system level must be remote. In civil transport aircraft this generally means the order of  $10^{-9}$  failures per hour in flights of up to ten hours.

One can immediately rule out some of the classical approaches to redundant systems on the grounds that they do not permit the detection and location of faults concurrent with critical operation. Other approaches can be dismissed because of insufficient redundancy and fault coverage. Still others are unusable because they depend excessively on the applications software.

The approach must have the ability to mask, i.e., correct, errors without requiring program rollback. All resources, including those used only in case of malfunction, must be capable of being individually verified during system operation. The approach must further be capable of surviving a multiplicity of faults, although not necessarily all at the same time.

Apparently, the most efficient way to furnish the multiple fault tolerance and concurrent testing is in a multiprocessing or multicomputing structure. Moreover, in order to provide error masking, all critical transactions must be at least triplicated. This is the course that has been followed in both the FTMP and the SIFT architectures. The result is a variant of classical redundancy of the TMR-Hybrid type [12], in which spare elements are placed in a pool so that they can substitute for any element in any of several parallel TMR triads. We find it convenient to refer to this redundancy

form as "parallel-hybrid" redundancy. Both FTMP and SIFT employ three times the resources nominally required by the application, plus an arbitrary level of spares, plus the hardware and software overhead necessary to manage the redundancy, i.e., fault detection and isolation, reconfiguration, and recovery. These two architectures employ graceful degradation as an important means of trading system cost against criticality. In projected aircraft, the flight critical functions account for a minority of the resource utilization. These functions are therefore supported with highest priority as resource pools diminish due to aggregated failures.

Beyond this point, FTMP and SIFT have gone separate ways. The FTMP has adopted a fully synchronous approach, which allows hardware-implemented bit-by-bit voting of all transactions. This in turn allows system management to be effected by majority rule, and means that the modules can be reassigned under executive control to different triads, or to spare status. Modules can be reconfigured in order to diagnose the location of a fault, to test the reconfiguration mechanisms, to activate spares for purposes of test and recovery, and to retire modules diagnosed as failed.

The next section discusses the theory of the FTMP architecture, and enlarges on several of the points that have been introduced here.

## II. THEORY OF THE FTMP

### A. Nominal Organization

Loosely defined, a multiprocessor is a computer with several processors and a single (possibly multiport) memory accessible to all processors. In the extreme, all instructions and data reside in a common memory available to any processor, so that processors are "anonymous." Given a suitable state vector, any processor can execute any procedure from any starting point. Motivations for multiprocessors are typically to increase productivity and availability at the same time, although these two purposes are competitive. At any rate, parallelism is intrinsic to the multiprocessor, as each processor is able to execute a different concurrent procedure subject to limitations imposed by resource sharing and sequential constraints on the procedures.

1) *Memory Access:* A "canonical form" of a multiprocessor is illustrated in Fig. 1, which introduces the notion of memory private to each processor in addition to the common memory. The rationale for this private, or cache, memory stems from the limitations imposed on parallel operation by memory access constraints. In a multiprocessor with highly parallel memory access, memory conflicts would occur only when individual units of data are simultaneously requested, or are locked for sequential conflict resolution. This would be the optimum structure for parallelism, and the cache memory's role is reduced to a possible enhancement of processor execution speed.

In the FTMP, on the other hand, the memory access is highly serial, for reasons dictated by reliability and economy. This essentially means that the memory has a single port, and that the throughput of the multiprocessor is governed by the bandwidth of this memory port. In this case, the cache memory has a significant role in enhancing parallelism. The combination of processor and cache is a true computer, capable of performing elaborate operations on input data in response to terse commands. This means that the common memory can contain programs written in a language level



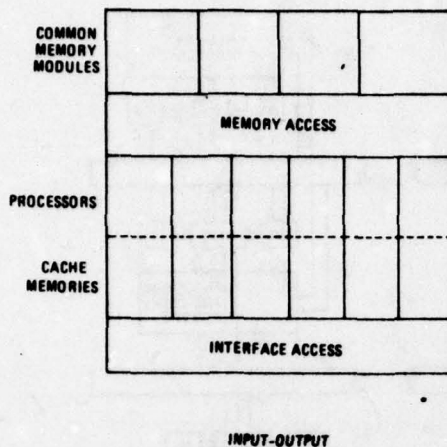


Fig. 1. Multiprocessor functional form.

higher than the processor's machine-language level, and that the processor-cache unit can interpret the higher level statements during the time that other processor-cache units are accessing the common memory. In this mode of system operation, which is really a form of "virtual machine," a memory port of moderate bandwidth can support an instruction execution "bandwidth" that is, at least in principle, almost arbitrarily large.

The degree to which the instruction execution bandwidth can exceed the common memory port bandwidth depends on the parameters of the cache memory, the terseness of the higher level language, and the relative amount of input and output data for each independent procedure. Clearly, the enlargement of the cache memories tends toward a multi-computer organization. Indeed, at some point the total cache capacity becomes adequate to contain everything in common memory, and the usefulness of common memory is reduced to the buffering of interprocess data. Processor anonymity is significant to this application because of the frequent reconfigurations that need to take place in this computer for latent fault exposure. Anonymity also provides an intrinsic mechanism for dynamic load distribution among available processing resources. The cache memory, however, acts to reduce the anonymity of the processor. To put it another way, the degree of anonymity is determined by the ease of reloading the cache memory. With zero cache memory, anonymity is greatest. As cache memory is increased to support instruction bandwidth enhancement, the anonymity of the processor-cache units depends on the amount of cache memory whose contents are unique to one processor. Note that the incorporation of identical procedural and other constant data, or indeed identical variable data, in every cache memory has no adverse impact on anonymity.

The use of a cache memory in a sampled-data control application, such as the aircraft application considered here, is generally productive. The typical job step uses rather few data samples as input, and produces one data sample as output. The procedures used tend to lend themselves well to expression as macrooperations, i.e., higher level operations, such as floating point arithmetic, linear combination, elementary functions, vector and matrix operations, and so forth. The incorporation of procedures of this level as

cache subroutines is reasonable and profitable in today's technology. The current high annual rate of memory density increase prompts one to observe that a fairly extensive set of procedures, and indeed a hierarchy of procedures, is increasingly appropriate for inclusion in cache memories.

The cache memory structure of the FTMP includes memories for data and procedures, partly read-write, partly read-only, designed to enhance instruction bandwidth with rather little loss of processor anonymity. The common memory, although highly modular, acts as a single-port paged memory, accessible to one processor at a time via a serial bus with a built-in contention mechanism.

2) *Functional Resource Allocation*: The programmer sees this multiprocessor as a machine for executing job steps, largely corresponding to periodic sampled-data updates. The magnitudes of these job steps will vary considerably from one control function to another, but will require something of the order of a few milliseconds, on the average, of processor time per job step. The procedure for each job step is written in a suitable language, and resides in common memory. Typically, each job step is scheduled to occur at a given time or following a given event. The relevant dispatch data for each scheduled job step is kept in a queue, where it is frequently examined to see if the job step is eligible to be run, or *invoked*. The frequent examinations are conducted by processors that have completed their earlier assignments, and are available to undertake new ones. When an available processor finds one or more eligible job steps, it selects one of them to invoke. In this way, job allocation is dynamic, and adjusts itself to the momentary load distribution and to module failures.

Input-output management in a multiprocessor can be more complex than it is in a single multiprogrammed computer, because as a single-port resource, it impinges on program parallelism. Depending on the statistics of external data traffic and of internal job steps, different access strategies may be appropriate. The most straightforward of these is to treat interface access as a single resource that is allocated to a single process for its exclusive use for the short period of time that a process requires access. Access may be granted on a priority basis or a first come first served basis. That is, when a processor needs interface access, it ascertains by means of flags in memory whether the interface is free. If not, the processor waits (with appropriate safeguards against lock-up) until it becomes free.

### B. Redundant Organization

The physical organization of the FTMP is substantially more complex than the nominal organization outlined in the preceding section. A simplified module diagram of the computer is shown in Fig. 2. Superficially, this diagram appears much the same as the nominal multiprocessor. The principal differences are that the buses for memory and interface access are redundant, and that the actual number of modules is three times the number of nominal modules plus some number of spares.

All activity is conducted by triads of modules and triads of buses. A module triad is formed by associating any three like modules with one another. This means that any module can serve as a spare for any triad. Such flexibility permits the best possible utilization of surviving modules. A single triad of bus lines is active at any one time for each of the

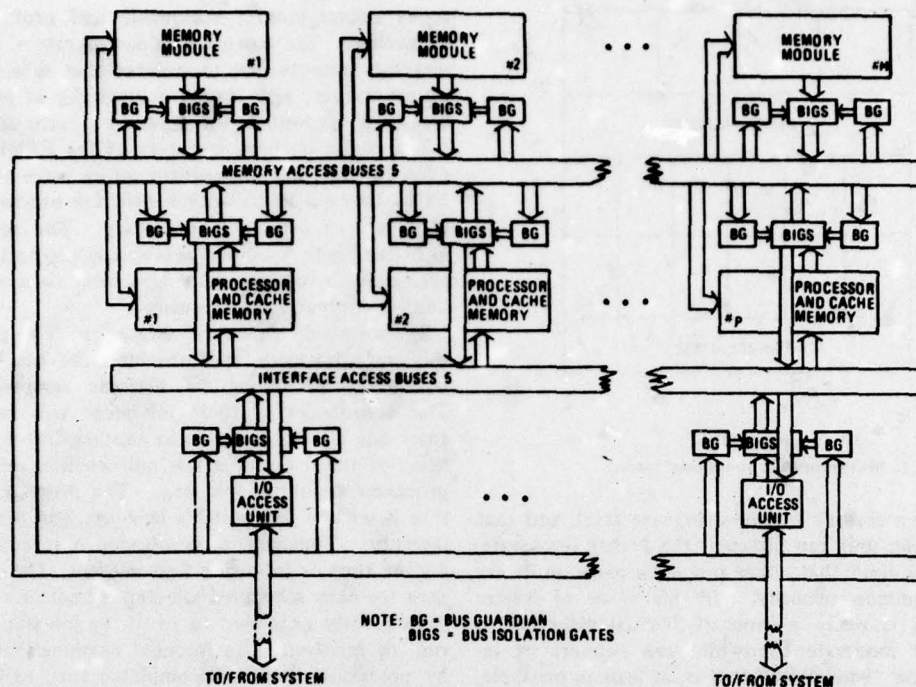


Fig. 2. Simplified physical diagram of the FTMP.

memory and interface accesses. In other words, a three-member subset of  $N$  bus lines is chosen on a quasistatic basis to serve as a bus triad.

Every module of every kind is able to receive data from all incident bus lines, and contains a decision element to formulate a corrected version of bus data. It is necessary for each module to know which three bus lines are the active ones. These three lines are connected to a voter in each module, thus constituting a TMR element. The three active bus lines carry three independently generated versions of the data, each version coming from a different member of the triad that is transmitting the data. To accomplish this, it is necessary to assign each module to transmit on one specific bus line. Now if totally flexible module configuration is to be possible, it follows that the assignment of a module's transmission to a single bus line must be quasi-static and reconfigurable.

1) *Bus Guardians*: In addition to the redundancy described in the preceding few paragraphs, the redundant organization differs from the nominal one by virtue of the inclusion of independent submodules called bus guardian units in each processor, memory, and input-output access unit. Guardians are charged with governing the status of their associated modules. This includes power-on status, memory bus triad and transmission selection, and certain self-test configuration selections.

Each of the functions of the guardian has the characteristic that its failure modes have safe directions as well as unsafe ones. By biasing the failure modes toward the safe directions, it is possible to increase the probability of system survival. In general, the safe failure modes of a module are power-off, and bus transmission disconnected. To bias in this direction, one can employ redundant guardians in each module, and require agreement among them to establish power-on and bus transmission enable.

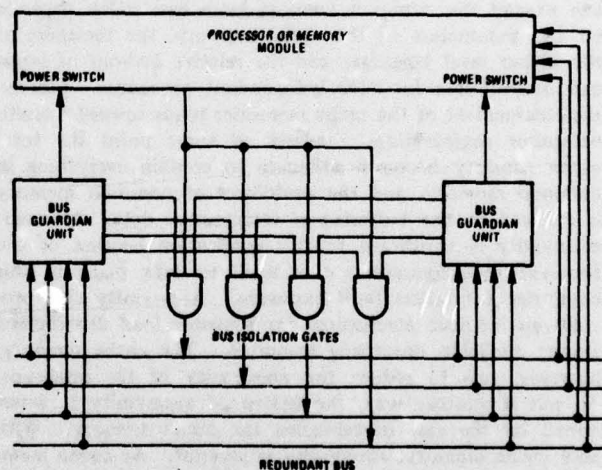


Fig. 3. Bus guardian connections.

The connection of bus guardians is illustrated in Fig. 3. It should first be noted that the guardian principle depends heavily on fault independence. Therefore, each guardian derives its power, its bus inputs, and its timing reference independently of all other guardians. It is moreover physically isolated from all other guardians and all modules. A particularly critical area from the isolation viewpoint is the control of the module's transmission interface onto the various bus lines. The bus isolation gates must be highly independent of one another, as must the guardian's enable signals to these gates. This is one of the crucial electrical and mechanical design aspects of the entire computer.

Bus guardians are addressable as part of the common memory address space, and are capable of receiving messages from any processor triad via the active memory bus triad. A mes-



sage to a guardian contains commands which are staticized by the guardian and applied to its outputs until superseded by a new command message. In this way, the probability is remote that a failed module can assert more than one erroneous data stream. As a result, correct data can be determined by the bus voters, and the malfunctioning module can be switched to a silent state. It is noted in passing that certain failures of a bus isolation gate can render a bus line useless, in which case the active bus triad must be reconfigured. However, most guardian failures are biased to appear as passive failures of the unit to which the particular guardian unit pertains.

Guardians are used as agents to convey the computer's configuration authority to all elements of the computer. They are highly secure against the random or willful malfunction of any single active transmitting module. They make possible the highly flexible reconfiguration on which the FTMP depends.

2) *Processor and Memory Modules*: All modules and buses are organized into triads. In the case of processors and memories, there can be numerous triads in existence at the same time, but only one memory bus triad and only one interface bus triad. Each processor triad acts as one functional processor, of which several can work in parallel. Each memory triad acts as a page of memory, of which several can exist at one time, but only one can communicate at a time with a processor triad.

When a processor fails, its triad will attempt to complete its current job step, which it will be able to do unless a second failure prevents it. The period of vulnerability to a second failure will be a fraction of a second. When the job step is complete, one of the other processor triads is assigned the task of reconfiguring the injured triad. When the erroneous module is identified, it is removed by commands to its guardians. If a spare is available, it is connected to the appropriate bus by its guardians, likewise upon command by the processor triad assigned to the reconfiguration. Triad identity will be assigned to the spare processor by a direct message. If no spares are available, the injured triad is retired. The resources of the multiprocessor are diminished by one processing unit, and the two unfailed members of the former triad are now available to be used as spares, should further failures occur.

The situation is much the same for memory modules. The principal difference is that memories are not anonymous. In fact, a read-only memory module is totally dedicated to its assigned function, and cannot be used as a spare. When a read-only memory triad is injured by the loss of a memory module, a read-write memory module can be used as a spare. It must be loaded to agree with the surviving triad members before a second failure occurs. If no spare is available, the triad is reduced to a dyad, which is vulnerable to the next failure, at which time one memory page is lost. This is a significant departure from the flexibility offered by the anonymous processor triads. The eventuality of read-only memory failure must clearly be covered by the inclusion of adequate spares, either read-write memories for flexible pooled use, or extra dedicated copies of read-only memory.

3) *Input-Output Access*: Fig. 2 indicates the existence of input-output access modules connected to the internal interface bus and also the external environment.

The external interfaces of the computer can alternatively support dedicated, bussed, or networked link structures to

the sensor and effector components. The redundancy structure at this point depends on the redundancy desired in the external interface.

The simplest conceptual structure is a triple-redundant interface, such as a redundant external bus, where the triple modular redundancy structure is extended through to the component interfaces. Each external bus line can be dedicated to a different input-output access module, which in turn is assigned by its guardian units to transmit on one of the active interface bus lines. More complex variants are possible, in which each access module performs error correction by voting on incoming data from the external bus.

When an external interface is nonredundant, the strategy would be to assign it to a single access module, where the module would transmit on all three active interface bus lines. A malfunctioning access module could pollute the entire interface bus, but with suitable encoding and protocol there would be no serious consequences to the state of the system. The offending access module could be discovered and disconnected by bus guardian commands conducted over the memory bus, the major penalty being a time loss on the remainder of the input-output interface of the computer. For dedicated links, the loss of the link is noncritical by hypothesis. For a network, whose survival is assumed critical [6], the computer must interface with the network in several places via several distinct access modules. Each such interface would be simplex, but the system would survive the failure of all but one of them.

### C. Synchronization

The employment of independent redundancy requires some form of synchronization among the independent data sources. Soft, or loose synchronization involves such operations as buffering, comparing or voting, signalling consensus, and marking completed intervals. These can be done by program, given suitable intermodule data links. Hard, or tight synchronization involves hardware comparison or voting, and a common time reference, whereas loose synchronization can employ separate time references.

Tight synchronization is employed in the FTMP. It provides the basis for solving some problems, and it presents some problems of its own. A common time reference, or clock, that supports hardware voting, allows instantaneous validation of internal data, configuration control, and, in some cases, interface data. In this way, it helps to make the redundant multiprocessor resemble the nominal one, which is advantageous to programmers at all levels.

The problems of common clocking stem primarily from the fact that it is critical to computer operation in the dynamic sense. The timing reference must be continuous and must remain within tolerances. A second consideration is that common clocking results in time-correlated data transfer, which is subject to correlated malfunction if subjected to external radiation of electromagnetic energy beyond the levels tolerated by shielding. The second problem is intrinsic to all synchronization, but is more severe for tight synchronization. The problem also exists in principle for any degree of shielding. When the statistics of such interference are known, the problem can be addressed in the time domain by encoding for error detection, rerun for recovery, or repetition for time independence.

The problem of maintaining a continuous timing reference is solved by a fault-tolerant redundant clocking arrangement,

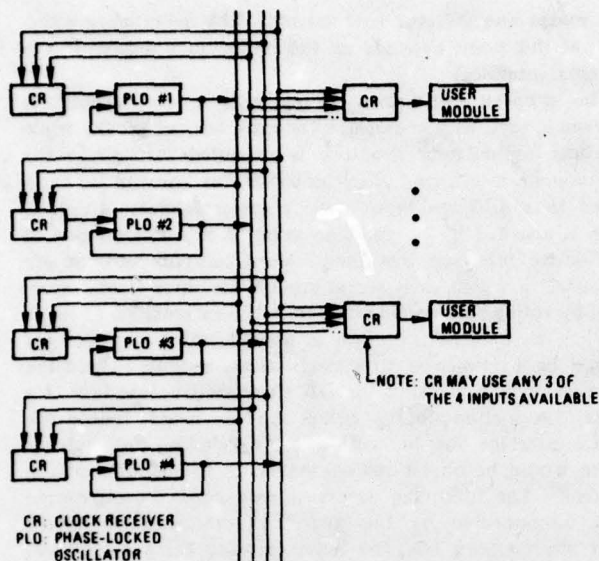


Fig. 4. Fault-tolerant clock system.

based on a majority logic algorithm described in reference [13]. A more recent embodiment, using voltage-controlled crystal oscillators, will be described in future reports. The basic principle of the system is shown in Fig. 4, which shows a set of independent phase-locked oscillators arranged so that the failure of one of the oscillators does not destroy the phase lock of the survivors. The clock signal from each oscillator is distributed to every module and guardian, so that each can make an independent determination of clocking edges. These independent determinations are made by circuits called clock receivers, whose operational principles are closely similar to the clock receivers described in [13]. In normal, nonfailed operation, the outputs of all the clock receivers are in phase lock with each other and with all the oscillators. The same phase lock holds when an oscillator fails. The failure of a clock distribution line appears as an oscillator failure, and the failure of a clock receiver appears as a failure of the module or guardian that contains it. The approach is discussed further in Section III-D6.

#### D. Malfunction Management

The unusually high level of dependability required in the FTMP makes it mandatory to consider all possible sources and effects of probable malfunctions. The probabilities associated with exposure to hazards are important here, as they are in any reliability analysis. The fact that reconfiguration and recovery are needed to meet reliability goals raises other issues of importance, having to do with the probabilities associated with the detection and identification of malfunctions, reconfiguration and recovery of the system, and the system status following a malfunction event. All those considerations relate both to the design and the evaluation of the system.

1) *Malfunction Sources:* A malfunction is a general term for anomalous behavior. Numerous kinds of malfunctions are distinguished, ranging from microscopic disorders in an integrated circuit to total aircraft impairment. Within the information processing segment of the total system, we

are concerned about avoiding malfunctions that preclude the availability of viable contingencies. We can think of potential malfunctions as being infinitely rich in number and variety, and tractable solely because they can be treated as classes and subclasses.

The first class of malfunctions to be examined is that resulting from externally induced phenomena, such as physical penetration, radiation (atomic, electromagnetic), temperature extremes, or excursion of prime power. The common thread in these diverse physical environments is that their effects can not be confined or localized to one or a few subportions of the information system. The entire system is vulnerable at one time, and for an arbitrarily high exposure it can not be made otherwise. That is, the shielding, structure, environmental control, and prime power generation must all be designed to withstand stated levels of exposure to known hazards. Exposures in excess of these levels are potentially catastrophic.

The second malfunction class is that of random malfunctions whose sources are internal to the system. Typically, these result from circuit failures. When idealized, such malfunctions are permanent, isolated, unambiguous, visible, and recoverable. Actual faults are apt to be marginal, intermittent, correlated, hidden, uncovered, and/or not perceived uniformly by multiple observers. This is the category of malfunctions that redundancy addresses, although the non-ideal attributes of actual faults tend to undermine the effectiveness of all redundant systems.

The third class of malfunction sources will simply be denoted as "other sources." The first two classes are broadly enough defined to be stretched to cover everything, but it is useful to emphasize certain sources separately. Thus we include in this third category the deficiencies resulting from lapses in system specification, that is, where the domain of operation and the domain of design are not matched. Software in this sense is a specification. It specifies the sequential rules of hardware utilization. Logic design is also a specification in this sense, as are design factors related to the human interfaces and the sensor and effector interfaces. The architectural implications of this category are that the system must be tractable and understandable enough to reduce the probability of occurrence of such malfunctions to a negligible level.

2) *Malfunction Consequences:* It has been useful to characterize the various possible malfunctions according to the levels at which they affect the system [14]. There are *physical malfunctions* that occur within hardware elements, such as a short circuit in a transistor. These have been referred to by various writers as faults and failures, and in this paper the word "failure" refers to this category. A physical malfunction may or may not result in a *logic malfunction*, in which a logic variable is at some time or another complementary to its correct value. Where authors use the word "fault" for physical malfunction, they use "failure" for logic malfunction, and vice versa. A logic malfunction can occur in the absence of a physical malfunction, notably from induced sources.

A logic malfunction may or may not produce a *data malfunction*, often called an error. A data malfunction can occur in the absence of a logic malfunction, notably from specification lapses. A data malfunction, in turn, may or may not produce a *subsystem malfunction*, which in turn may or may not produce *system malfunction*.



We have portrayed a propagation chain from physical malfunctions to system malfunction, with some external entry points. Whether propagation takes place from one level to another depends on whether a causal link exists in the first place, and whether the phenomenon is masked by a redundancy. Thus a logic malfunction produces a data malfunction only if it impacts the outcome of an operation. Even then, it may not, as for example when the data results from the voting of three inputs, only one of which suffers a data malfunction.

A key point, often overlooked in simplistic treatments of redundancy, is that redundancy always has a limited capacity to mask malfunctions, and this capacity can degrade to zero without affecting the apparent behavior of the system. Therefore, a system designed to have tolerance may in fact have none at the inception of a critical mission. Alternatively it may have some tolerance, but less than the design level, and less than what is assumed. Masking is a two-edged sword. On one hand it is a mechanism for holding malfunctions at a low system level, while on the other hand it may obscure the fact that the malfunction has occurred and thereby has reduced the system's tolerance to future malfunctions [15].

3) *Tolerance Renewal Principles*: The primary advantage of hybrid redundancy over TMR is that injured triads are reconfigured back to a state where they can once again mask malfunctions. This is a process of tolerance renewal. In principle, the system failure rate is restored to its design value by the reconfiguration process. If reconfiguration were to fail, the system failure rate would increase, possibly by many orders of magnitude.

In practice, there are several ways in which an injured triad can fail to be reconfigured. These include exhaustion of spare modules, malfunction of the reconfiguration mechanism, failure to detect the need to reconfigure, and perhaps the use of a defective spare module. We can characterize the process of tolerance renewal as the detection and location of any physical malfunction, the removal of vulnerability from the triad containing the malfunction, the replacement, by spares, of functions thus removed, and the initialization of the reconstituted triad. All mechanisms involved in this process are subject to malfunction, of course, and such malfunctions constitute injury to their triads, and require that tolerance renewal be carried out on the appropriate modules.

The tolerance renewal mechanism in the FTMP is largely contained in the voters and the bus guardian units. Both the voters and the guardian units possess bus line interfaces, and therefore are both capable of degrading elements (i.e., bus lines) outside of their own modules (e.g., processor, memory, interface access). This by itself is not qualitatively different from a single malfunction. The important concern is that all guardians in a single module may fail in such a way as to enable that module to transmit on more than one bus line. Design steps are taken to minimize the probability of this eventuality, but the probability is finite that it will happen. A subsequent failure of the module in a malevolent state could cause an entire central computer to malfunction.

4) *Fault Detection, Identification, and Recovery*: The FTMP is designed to have a highly improbable loss of capability, with a total failure rate of less than  $10^{-9}$  failures per hour in a flight of up to ten hours. This virtually rules out the use of ordinary triple modular redundancy, as the MTBF's achievable in large scale production have been consistently too low for such reliability without replacement

of failed modules. Therefore some form of hybrid redundancy is needed. In a simplistic view, hybrid redundancy works by substituting a spare the first time the TMR voters disagree. This view has the shortcoming of not taking latency of faults into account. That is, the first fault may not result in any voter disagreements, whereas when combined with a second fault, it may frustrate recovery. A prerequisite for achieving highly improbable failure in a hybrid system is therefore to expose latent faults by systematic exercising, or "flexing" of all logic elements. The flexing period must be of the order of seconds for a reasonably sized system with module MTBF's in the ten-thousand hour range. Clearly, then, flexing cannot be relegated to preflight checkout, but must rather be conducted routinely in flight. An ordinary hybrid TMR system cannot routinely test itself when performing critical functions, as it is vulnerable during these times. A parallel hybrid TMR system can do this, however, and this becomes an integral part of the computer's architecture.

In the FTMP, an error correction mechanism exists in every module in the form of a voter. Each voter must be tested routinely to ensure that its error correcting capability is undiminished. Bus voters under normal conditions will correct single bus errors and will set error latches to indicate which of the buses was in disagreement. At this time the processor can record the identity of the nominal user of the bus for diagnostic purposes. A processor triad can flex its own voters during a test job step by having each triad member purposely utter independent bus data that causes all possible kinds of bus errors. To pass the test, all triad members must receive the same data, form the same corrected result, and indicate the same disagreement patterns in their error latches. This is a relatively simple test procedure, which can be conducted by a processor triad under test while other triads carry on normal functions. In a sense it qualifies the triad to conduct further testing, in which the triad's voters are the decision elements.

The remainder of the system testing function is carried out under the assumption that the processor voters and error latches are operational. The test process involves the conversion of every fault into an error, by making calculations whose results are sensitive to each logic variable. Each bus and module, including voters, guardians, isolation gates, clock receivers, oscillators, and data and power interfaces must be exercised in depth.

We might summarize the fault detection process as the arrival of disagreement errors at the voters of a processor triad, stimulated by normal or test activity. The detection of a fault initiates the process of fault identification, which is the discovery of the module, bus, or other isolated element in which the failure resides. During the testing process for latent faults, there is relatively little ambiguity in the determination of faulty modules. In normal operation, however, an error on the bus can come from a number of sources. The identification of the faulty module generally requires the "rounding up of suspects," that is, the listing of elements that transmit on the disagreeing bus. If a module fault is permanent, the module can be found by moving it to another bus. If the bus is faulty, reconfiguration will not move the error to another bus.

Intermittent faults are less easy to identify. When the source of an error eludes detection by disappearing, all of the suspect elements are assigned one demerit, and a recon-

figuration is then made to distribute the suspects evenly on different buses. Subsequent error occurrences and reconfigurations will cause a preponderance of demerits to accumulate in the name of the faulty module or bus.

The recovery process is one of assignment and initialization for modules, and voter and transmitter selection for buses. These are all accomplished by the bus guardian units upon receipt of commands from active triads executing system software. Recovery can take place even if single errors are present on the buses. In principle, therefore, an injured processor triad can reconfigure itself.

The use of program restart, or rollback, as a recovery mechanism is secondary, because it is neither sufficiently effective nor easy to implement. The first level of system defense is the masking of errors by the TMR method. The additional system failure rate reduction achievable by rollback can not be measured, *a priori*, without an understanding of the applications software. It should be anticipated, however, that any event that defeats the TMR masking is apt to destroy the vehicle's state vector, which may or may not be catastrophic. In any event, some degree of program rerun should be included to support power-up initialization and to deal to some extent with the eventuality of uncovered errors. This will affect both system software and application software.

### III. DESCRIPTION OF AN ENGINEERING PROTOTYPE OF THE FTMP

During the 1978 and 1979 time frame the Charles Stark Draper Laboratory is planning the construction, for NASA, of an engineering prototype of the FTMP. The hardware is to be built by a major avionics manufacturer using specifications provided by CSDL. CSDL will retain program responsibility, provide all system software, and will conduct the integration, test, and evaluation of the system. The project is being sponsored by the NASA Langley Research Center as a part of the Energy Efficient Aircraft Program. The implementation of the prototype is discussed in this Section.

The proposed system is to be constructed of ten identical line replaceable units (LRU's) connected as indicated in Fig. 5. Each LRU contains one processor/cache module, one memory module, one I/O port, one clock generator, and related peripheral support and control circuitry. Fig. 6 shows how an LRU is divided into fault-containment regions. The principal region is detailed in Fig. 7.

Up to three processor triads can be in operation simultaneously, utilizing nine of ten available processor/cache modules. The tenth module serves as a spare. With three triads operating simultaneously, the system is functioning as a three-processor multiprocessor.

Up to three memory triads can be formed from nine of the mass memory modules. The tenth module is a spare. Each memory triad is assigned to service a single 16k work region of the shared mass memory address space. With three memory triads operating simultaneously, 48k words of contiguous shared mass memory address space can be serviced.

The I/O ports use MIL-STD-1553 data formats and signalling protocols. MIL-STD-1553 is a United States Air Force standard for a bit serial, time multiplexed avionics data bus. A single I/O port accepts the bit serial data from a processor triad, votes to mask any errors in that triad, and generates a single version of the I/O transmission. This version is electrically transformed to conform with MIL-STD-1553 specifications, and is transmitted to the outside world

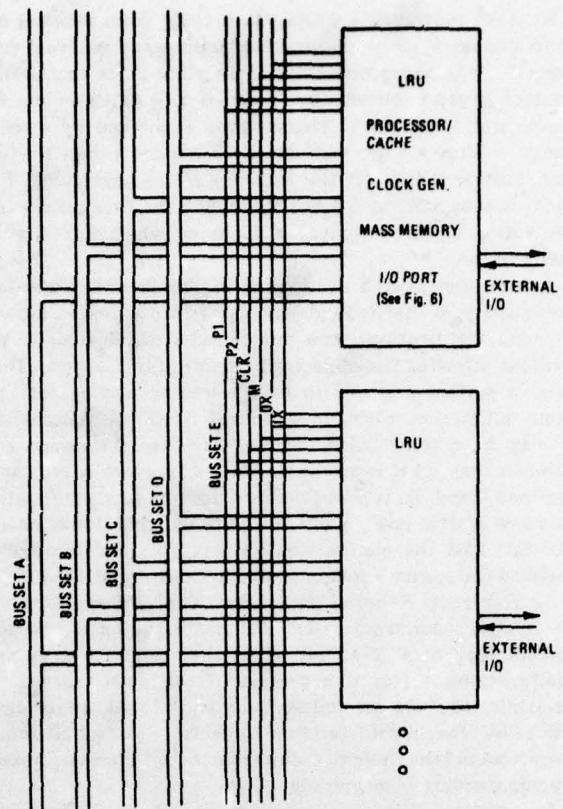


Fig. 5. LRU and Bus interconnections.

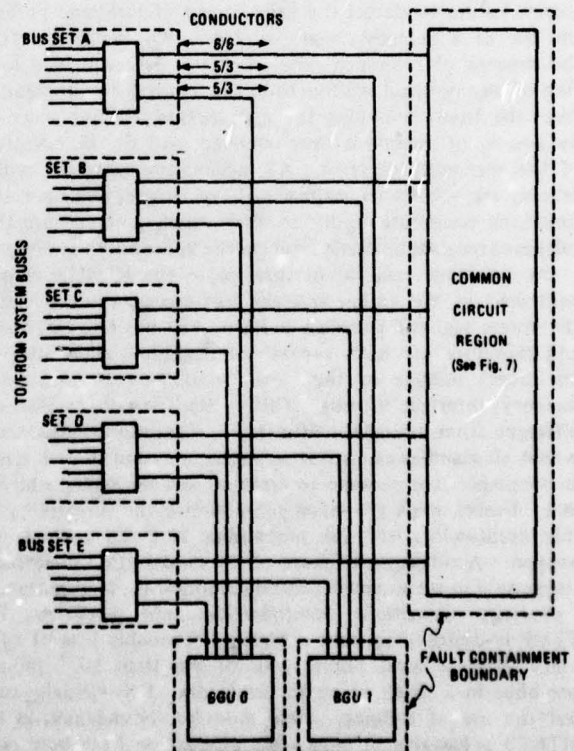


Fig. 6. LRU fault containment boundaries.



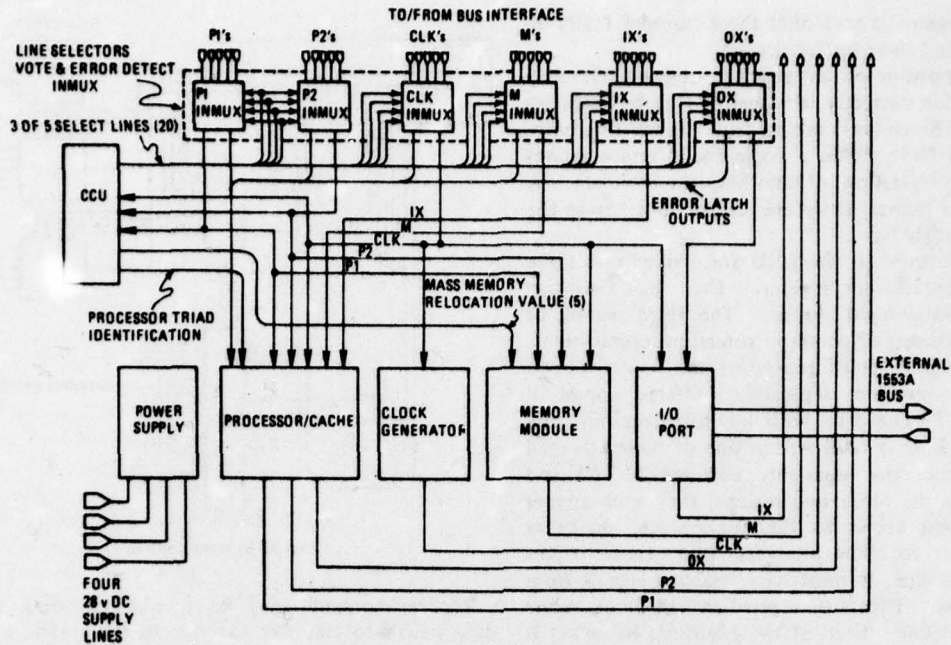


Fig. 7. Common circuitry region.

on one member of a full-duplex transmission pair. Received data from this MIL-STD-1553 transmission pair is accepted by the I/O port, converted to an internal signal level, and distributed to all processors. At least one port and its associated external transmission pair must remain functional for the system to remain operational. Error detection and correction outside the multiprocessor relies upon data encoding and time redundancy in communications to and from remote terminals.

This engineering prototype differs from the basic FTMP design in that it groups a processor, a memory unit, and an I/O port together in a single LRU with common power supply, bus guardians, isolation gates, and other common-failure elements. The reason for doing this arises from the physical form factors involved. Meanwhile, this design preserves the necessary features to allow processors, memory units, and I/O ports to be assigned independently of each other, and for the system to diagnose and recover from simultaneous failures of all three.

#### A. The Redundant Bus Structure

The bus system shown in Fig. 5 is quintuple-redundant. Each bus has lines dedicated to processor transmission, (the two *P* bus lines); memory module transmissions, (the *M* line); clock generator transmission, (the *CLK* line); and I/O transmissions, (the *IX* and *OX* lines). Subsets of three of the five buses are assigned to carry processor and memory triad data. A subset of four of the five is used to carry clock generator transmissions. A single bus of the five is used to carry I/O port transmissions.

The processor uses two bus lines, *P1* and *P2*, to transmit data and commands to common memory and status register devices. The processor triads also contend for control of the bus system via a cooperative, competitive allocation technique which uses these bus lines.

A triad of memory modules uses the memory bus lines to transmit data requested by a processor triad. Since memory triads only speak on command, there is no mechanism, such as the competitive poll used by the processors, to grant permission to transmit. The processor in control of the *P* bus implicitly grants transmission permission by issuing a read request.

#### B. LRU Interfacing to the Bus System

Each LRU of the system must be interfaced to the bus system in a fashion that protects the fault-tolerant architectural features of the logical design. Several design constraints must be met in order to meet this requirement. Fig. 6 illustrates a suitable interface.

Each of the five buses is connected to the LRU through a dedicated bus interface. Each of these bus interfaces represents an independent fault containment region. Design requirements for a fault containment region limit the physical impact of a fault to that region. Signal lines into and out of the region are buffered at the region's edge so that a fault on any of these lines external to the region will not affect the correct operation of the circuitry within the region, excepting possibly these output of input buffers. The principal concept of a fault containment region is the containment of physical damage to one region by the surrounding regions. The logical containment of the effects of a fault are provided by other means. For example, a fault such as a short circuit to power on all lines into and out of a bus interface has two partitionable effects. First, data transmitted through that bus interface is likely to be received incorrectly. This is the logical impact of the fault. The logical failure is not contained by the fault containment region. The second effect is physical. The fault will electrically stress the receiving and transmitting buffers of attached regions. This stress may induce physical faults within these buffers, but the

design of these regions is such that these internal faults do not propagate beyond these buffer circuits.

The remaining portion of an attached region's circuitry continues to function correctly, although it may be operating on incorrect data. Since there are no fault propagation paths between regions, a fault within a single bus interface cannot affect the correct operation of another bus interface. A single bus interface failure, therefore, can at most cause the apparent loss of a single bus.

The remaining portions of the LRU are divided into three additional fault-containment regions. Each Bus Guardian Unit is a fault-containment region. The third region, or *principal region*, consists of common voters, processor/cache, mass memory, I/O port, clock generator, and power supply. The bus interface provides separately buffered copies of the *P1*, *P2* and *CLK* lines to both bus guardians and the principal region. Since a fault within one of these attached regions cannot affect the separately buffered *P1*, *P2* and *CLK* lines used by the other two regions, they each appear to have independent access to the bus system. In order for a bus interface to allow principal region transmissions onto a system bus line, it must have enabling signals from both bus guardians. Thus either guardian can block access to a particular bus line. Each of the guardians has what is effectively independent access to all incoming bus data. It can independently mask single bus errors via voting, and it processes incoming processor triad transmissions, responding only to write commands to its particular address location. The contents of these write commands alter the static enabling signals from the guardians. Each guardian provides an enable line to each bus interface for the *P* lines, *M* line, *CLK* line, and *OX* line.

The LRU interfacing is designed to protect the integrity of the bus system despite multiple sequential faults. A worst case bus interface failure can at most disable all of the lines of only one of the quintuple bus sets. The system can then be reconfigured to use the remaining lines of other buses. One element of a triad or the clock quad, if it fails, can impact at most one of the active bus sets. Again, reconfiguration commands can isolate that faulty unit from the bus and assign a spare to replace it, thereby restoring system health. To cause a system failure, four of the five bus sets must fail, or two bus guardians within the same LRU must fail, enabling the principal region to access all bus lines, and in addition, the principal region must fail.

#### C. System Control Units

The bus guardian unit is a particular case of a generalized unit called a *system control unit*. Each LRU has four system control units. They are designated bus guardian unit 0 (BGU 0); bus guardian unit 1 (BGU 1); configuration control unit (CCU); and the interprocessor triad communication unit (IPC unit). The CCU and IPC units are part of the principal fault containment region. As previously stated, BGU 0 and BGU 1 are each a fault containment region.

All of these system control unit types are similar and can be constructed from the same circuit. Fig. 8 illustrates the functional requirements for such a common circuit. Essentially the circuit must take the serial processor command data, *P1*, *P2*, and *CLK*, pass it through error-correlation circuitry, if this data is in redundant form, and convert it to a parallel form. A system control unit only responds to a memory write command to its own particular memory address.

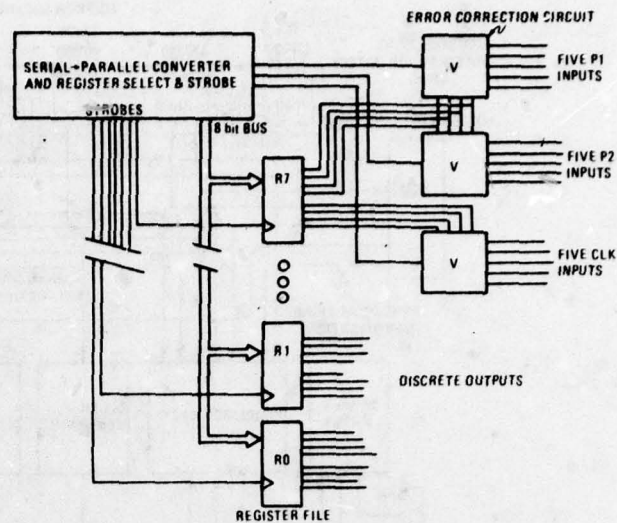


Fig. 8. System control unit.

Register contents may be supplied as static enabling or data signals to circuitry external to the system control unit, or they may be used internally to control the error correction circuitry (if present).

A power monitoring circuit switches the register store to battery power when primary power to the unit is not within specification. When battery powered, the register contents are protected, and the enabling lines from the guardians are in the disable state. Total loss of all power to a guardian clears the register contents to the disable state.

#### D. The Principal Fault-Containment Region

All of the circuitry of an LRU is within the bounds of the principal fault-containment region excepting the two bus guardians and the bus interfaces.

The principal region can be viewed as being made up of seven subregions. These are: 1) input processing; 2) configuration control; 3) processor/cache; 4) memory; 5) I/O ports; 6) clock generator; and 7) power supply, as shown in Fig. 7.

1) *Input Processing*: All input to the principal region is from the bus interfaces, and is first processed by shared signal selectors, voters, and error detection circuits. The input circuitry generates a single version of the *P1*, *P2*, *CLK*, *M*, *IX*, and *OX* lines to be used by all modules within the region. This single version of each line is the appropriate majority function of the selected group of 3 out of 5 lines. Additionally, the voting circuitry detects and latches any error condition on the bus lines, and provides this information as input discretes to the processor. The selection of one of the ten possible groups of 3 out of 5 buses to be used by the majority circuitry or the selection of which *IX* line to be used, is made by selector discretes provided by the Configuration Control Unit.

2) *Configuration Control Unit (CCU)*: The Configuration Control Unit (CCU) is a system control unit. The CCU is used to control the INMUX circuitry, is used to assign the processor/cache unit to a processor triad and to start and stop the processor, and is used to assign the mass memory module to a memory triad.

3) *Processor/Cache Module*: The processor/cache memory module is the most complex of the principal region. It can be partitioned into a number of submodules. These are: a) pro-



cessor, b) cache memory, c) bus controller, d) IPC unit, and e) MIL-STD-1553 controller.

a) *The processor:* The principal design requirements of the processor could be met using any of a large number of general purpose 16-bit minicomputer architectures. In order to support the projected computational requirements of an integrated avionics system, the basic processor has a raw instruction execution rate roughly equivalent to 500 000 16-bit fixed-point adds per second. A 16 bit fixed-point multiply has an execution time six times that of the fixed point add.

The instruction set of the processor is suitable for avionics applications and, in addition, provides for the following: 1) code is relocatable without modification; 2) Code is read-only and reentrant; 3) the CALL and RETURN instructions support dynamic program loading efficiently; 4) memory protect is supported for a region of the cache RAM; and 5) privileged user modes of operation are provided to prevent the direct execution of I/O and mass memory access instructions by applications code.

The processor is adapted to use the output of the CLK generator as its time base and incorporates a microcode interlock with the bus controller which allows three processors to be synchronized by using particular bus events, such as bus grant.

b) *Cache memory:* The cache memory is a 4k X 16 semiconductor RAM and 4k X 16 semiconductor PROM array. It interfaces to the processor over the processor's internal parallel bus. Access time for this memory is 400 ns. There is no requirement for nonvolatility in the RAM portion of this memory.

c) *Bus controller:* The bus controller is responsible for the bit-by-bit control of the processor side of bus activity. On command of the processor, the bus controller conducts a competitive polling sequence to acquire control of the main memory bus. The controller then holds the bus until instructed to release it. It makes use of the triad identification provided by the CCU and a priority field provided by the processor during the polling sequence. While holding the bus, it performs memory reads and writes as requested by the processor. Data and memory address transfers between the processor and controller are handled in parallel. The controller performs the necessary timing, serial to parallel and parallel to serial conversions for the processor. The processor handles block transfers performing the necessary housekeeping, streaming parallel memory addresses, and accepting whole word data streams from the controller and storing them in cache memory, or streaming parallel addresses and data to the controller for storage in the common memory.

d) *Interprocessor triad communication unit:* The Interprocessor Triad Communication Unit (IPC) is used by the executive for direct processor-triad to processor-triad communications. The IPC registers are available as discretes to the processor.

e) *MIL-STD-1553 controller:* A MIL-STD-1553 controller interfaces to the processor over the processor's internal parallel bus. It conforms to the standard format, except that the outgoing and incoming data paths have been split so as to provide full-duplex transmission paths.

4) *Memory Module:* The memory module contains a 16K X 16 CMOS memory array with the appropriate control circuitry to respond to processor triad memory read and write commands.

Input to the memory control circuitry is the bit-serial quantity represented by the outputs of the P-INMUX outputs and CLK-INMUX. The most significant bits of the incoming address are compared to the relocation register provided by the CCU. If they match, a read or write operation is performed. If they do not match, the incoming command is ignored. Read responses are made using the M bus. Responses are clocked using the output of the CLK-INMUX.

5) *I/O Port:* The I/O port is principally a signal level shifter and data synchronizer. A single corrected version of I/O output data, OX, is accepted by the I/O port from the common input module, and is buffered to conform to MIL-STD-1553 specifications. The transmitting processor triad is responsible for formatting the OX lines signal to conform to the MIL-STD-1553 format.

The I/O port receives I/O input data, synchronizes it so that transitions do not occur near system clock edges, converts the signal levels to an internal standard, and transmits the signal on an IX line to all processors.

6) *Clock Generator:* As discussed in Section II-C, the entire fault-tolerant multiprocessor rests on an assumption of synchronized operation based on a common timing reference. Each LRU includes a clock generator which can be synchronized to the common reference, and which, if gated by the BGU's onto a CLK bus, could serve as a contributing element to the common reference in the manner shown in Fig. 4. The clock generation circuit of an LRU interacts with the CLK bus lines, the CLK-INMUX, and the other clock generators. To understand the function of the clocking system, it is necessary to discuss all of these components as they interrelate with one another.

The clock bus is a component part of the quintuple redundant busing system. Each of the five bus sets includes one clock bus line, CLK. Normally, four of the five CLK lines are active and one is inactive. Four clock generators are chosen as the clock sources, each being assigned to a different clock bus. Each transmits a clock signal which is phase-locked to the other three active clock generators. Thus the system has available at all points a quad-redundant time base. Each clock receiver listens to three of the four active clock buses and generates a derived clock which remains correct even if one of the three input signals fails. It is therefore possible to tolerate a single failure of one of the elements of the clock quad without affecting the correctness of the derived clocks generated throughout the system.

Each bus guardian and each CLK-INMUX uses a clock receiver to generate its own corrected version of the system clock, despite single faults in the clock quad.

Each clock generator, whether active or in standby mode, phase locks its output to its CLK-INMUX output. Thus the clock generator outputs a clock which is in phase with the majority of three CLK buses. When active, the output of the clock generator is gated onto one of the four CLK buses, and its associated CLK-INMUX is adjusted to listen to the other three CLK buses. In this configuration the correctly functioning clock generators will produce multiple phase-locked clocks which will remain phase-locked despite any failure of a single clock element of the quad.

When a failure is detected, the system reconfigures, replacing the failed CLK bus or clock generator. Standby clock generators are already phase-locked to the corrected system clock, so that they can be switched in to replace a failed

clock generator with minimal transients in clock frequency and with negligible risks. This restores the fault-tolerant character of the clocking system, positioning it to tolerate the next clocking component failure.

7) *Power Supply*: The power supply provides regulated power to the LRU. The power supply can draw power from any of the four primary 28-V dc power buses. A circuit breaker or fuse protects each of these buses from a short circuit within the LRU. The power supply must have adequate energy storage so that its output remains within regulation for the time it takes these protective devices to act and the bus voltages to return to normal after a short circuit within another LRU. The output of the power supply is overvoltage protected, possibly with serial redundant protection.

The bus interface devices will be designed to operate safely for all power supply voltages beneath the overvoltage protection limit; that is, the bus interface will present a high impedance load on the bus for all voltage levels if the corresponding enables from the BGU's are unasserted.

The BGU's will monitor power supply voltages. If out-of-regulation voltages are detected, the contents of the BGU registers will be frozen, and all enabling outputs will revert to the unasserted state.

A battery backup is used to provide power to the CMOS memory array, and to the BGU and CCU register files, when primary power is lost. If this battery power fails when primary power is down, the register files of the BGU's and CCU will be cleared.

#### E. Primary Power

Power is distributed to all LRU's of the system by means of four 28-V dc power buses. Four 400-Hz 110-V dc to 28-V dc power converters provide power to these buses. These power supplies are overvoltage and overcurrent protected. If an overcurrent condition arises, the 28-V dc output will current-limit but return to normal when the protective devices within the shorting LRU open. Energy storage with the power supply must be adequate to tolerate momentary power interruptions such as are typically caused by power switching in aircraft power distribution systems.

### IV. SURVIVAL AND DISPATCH PROBABILITY MODELS FOR THE FTMP

The FTMP has several different failure modes, each of which is amenable to a different mathematical tool. Specifically, the probability of failure due to exhaustion of spares can be adequately modeled using combinatorial methods, whereas Markov processes are better suited to modeling coverage-related problems. Fortunately, each of these failure modes predominates in a different time segment, and therefore can be modeled and analyzed independently.

#### A. Survival Probability Models

The computation of survival probability of the FTMP for random hard failures is divided into the following three phases:

- 1) probability of failure due to the lack of perfect coverage using a Markov process model;
- 2) probability of failure due to exhaustion of spares using a combinatorial model;
- 3) probability of failure due to BGU failures in enable mode using a combinatorial model.

In the FTMP some time is required to detect, isolate, and recover from any failure. During this time a second failure may arrive in such a place as to be catastrophic. Therefore, the coverage [16] is imperfect. This phenomenon is most conveniently modeled using Markov processes, as each distinct failure or recovery moves the system into a state that is dependent only on the present state of the system. However, to limit the number of states to a reasonable level, it is necessary to make some approximations. The most effective of these approximations is to assume that recovery from a failure returns the system to a perfect state, which is the initial state of the system, rather than to a computationally degraded state. In effect, this implies an unlimited supply of spare units of each kind. The probability of failure due solely to exhaustion of equipment can be computed independently using combinatorial methods. The basic premise which allows one to decouple and model these two modes of failures separately is the predominance of each mode during a different time span. As will be shown in the following sections, in the short run (0-50 hr) it is the threat of near simultaneous failures which most affects system survivability, whereas in the long run (>100 hr) the system is likely to fail due to a lack of equipment. In addition to these, there is a third failure mode peculiar to the FTMP architecture that has to be accounted for. This relates to two bus guardian units in an LRU failing so as to enable a failed unit (processor, memory, etc.) to transmit simultaneously on a number of buses. It will be shown that this mode does not affect the reliability since its probability is insignificant at all times.

The following three subsections describe the models and the results.

1) *Lack of Coverage: Markov Model*: Since all the information as well as all the computations in the FTMP computer are triply redundant, any single failure in the system is completely masked by the majority voters. Therefore, if the system starts out in a totally fault-free state, it takes at least two successive failures without recovery to produce a catastrophic system failure. However, not all double failures are catastrophic. In fact, most double failures can be tolerated by the FTMP without any problem. The following is a list of all the catastrophic double failure combinations:

- 1) two processors in a triad fail;
- 2) two memory modules in a triad fail;
- 3) two active buses fail;
- 4) one active bus fails and a processor or memory enabled on another active bus fails;
- 5) two active oscillators fail;
- 6) one active bus fails and an oscillator enabled on another bus fails;
- 7) one LRU fails in common mode and an associated processor, memory, or bus fails;
- 8) two associated LRU's fail in common mode.

The common mode LRU failure refers to a failure of any of the LRU components that are shared by the processor, memory, and I/O port in that LRU. These include the local power supply, the oscillator, the two BGU's, and the selectors and voters. A local power supply failure in an LRU, for example, will result in the simultaneous loss of the processor, memory, and I/O port in that LRU. The BGU failures include only the disable mode, since the enable mode is taken care of separately. Finally, the bus failure includes a failure of any of



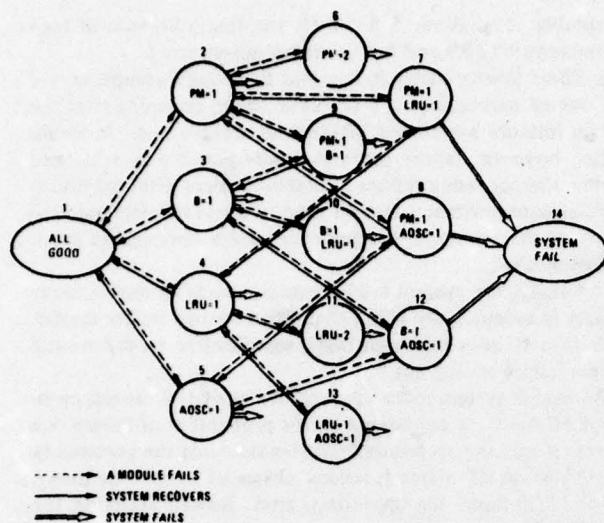


Fig. 9. Reliability model for lack of coverage.

the five lines constituting a bus or a failure of any of the ten bus interface gates connected to that bus.

A Markov model of the FTMP computer reliability based on the above discussion is shown in Fig. 9. The system is initially in a completely fault-free state or "ALL GOOD" state. It will be shown shortly that at time  $t = 0$ , such as a take-off time, the probability of having a latent failure in the system should be about  $10^{-6}$  to achieve a system failure rate of  $10^{-9}$  failures per hour. That is, one must be certain with a probability of about 0.999999 that the system is initially fault-free. In the following discussion, it is assumed that the system is initially fault-free. Some of the other assumptions used in developing the model are outlined below.

As explained earlier, it is assumed that reconfiguration around a failed unit returns the system to the perfect state. It is also assumed that all the failed buses are active and that all triple undetected faults cause system failure. These simplifying assumptions reduce the number of states in the model considerably without significantly altering the system failure probability. For example, contribution of triple faults to the system failure probability is found to be less than two per cent.

A baseline set of failure and recovery rates, as shown in Table I, was used to obtain a numerical solution of the Markov model. The values shown in Table I are the mean values. The model uses random values that are exponentially distributed around these means. One may argue about the fidelity of exponential distributions, although it is our contention that they represent the actual reconfiguration time distributions sufficiently well for this purpose [17].

The results of the Markov model are shown in Fig. 10 by the curve labelled "lack of coverage." It shows the system failure probability as a function of time on a log-log scale for the baseline hazard and recovery rates. The failure probability is seen to be a linear function of time (linear and unity slope on the log-log graph) which can be explained as follows. After an initial transient, which may take several hundred seconds to settle down, the state probabilities for all states except the system fail state become nearly constant. During this equilibrium there is a constant leakage of probability into the trapping state since all the transition rates are time invariant.

TABLE I  
BASELINE PARAMETER VALUES

SYSTEM CONF.		CYC	FAILURE RATE (PER HOUR)	MTBF (HRS)	RECOVERY TIME (SEC)
# PROCESSORS	10	5	$2 \times 10^{-4}$	5,000	0.25
# MEMORY UNITS	10	2	$2 \times 10^{-4}$	5,000	0.25
# IO UNITS	10	1	$5 \times 10^{-5}$	20,000	
# BUSES	5	3	$10^{-5}$	100,000	0.25
# MAIN POWER SUPPLY UNITS	4	1	$10^{-4}$	10,000	
# BGUs	20		EN = $10^{-6}$ DIS = $10^{-5}$	1,000 100,000	
# LRUs	10		CNF = $1.46 \times 10^{-4}$	7,000	0.25
# OSCILLATORS	10	3	$10^{-5}$	100,000	1.0

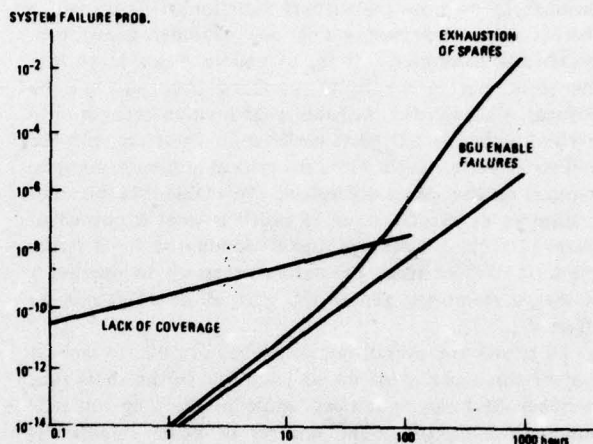


Fig. 10. System failure probability.

Since the total leakage rate is only about  $10^{-9}$  per hour, the state probabilities diminish extremely slowly, and a state of equilibrium would hold for hundreds of hours. For the baseline case, the system failure rate due to lack of coverage is found to be about  $3 \times 10^{-10}$  per hour.

The reason for having an initial latent-failure probability of  $10^{-6}$  now becomes clear. This is the probability of the system being in states 2 through 5, that is, the single-undetected-failure states (see Fig. 9). The transition rate from those four states into the system fail state or the probability of arrival of a second catastrophic failure is of the order of magnitude of  $10^{-3}$  per hour. To prove that the system is initially fault-free with absolute certainty is not possible. The triple redundancy prevalent in the system immediately points to any obvious disagreements and component failures, and a systematic exercise of all parts of the system using diagnostic routines can uncover most undetected faults. But this still leaves some types of faults, such as pattern sensitive memory locations, which can not be uncovered without exhaustive testing. The probability of such latent failures, has to be reduced to an insignificant level.

2) *Exhaustion of Spares Combinatorial Model:* In order to compute the probability of not having sufficient equipment, it is necessary to define the minimum equipment necessary to operate successfully. This is mission dependent as well as architecture dependent. The minimum equipment required to fly an aircraft shall be denoted as the Critical Minimum

Complement (CMC). The architecture-dependent parameters of the CMC include the power supply units and buses. One main power supply unit is deemed sufficient to run the whole computer. Similarly, two buses are adequate at the minimum to support communication between processors and memories, as well as the distribution of the clock. However, for one pathological clock failure mode it would be necessary to have three buses. The minimum number of processors and memories required is mission dependent. The throughput of the FTMP computer in a fully operational state is estimated to be 500 000 operations per second and the minimum throughput necessary to support all flight-critical functions is estimated to be about 200 000 operations per second. Similarly, the total storage capacity of the computer is 48 000 words while the critical programs are estimated to be less than 16 000 words. Thus two processor triads and one memory triad have to be operational to support the critical functions. There are a number of ways of achieving this, one of which uses 5 processors and 2 memories. It is, of course, possible to lose another processor in the fully populated triad and still be operational, although the probability of such an event is only 3/5. The number of I/O ports necessary to interface with the I/O network is one. Table I lists the critical minimum complement based on the above discussion. This table lists the minimum number of oscillators as 3, which is what is needed to generate a clock. However, this is dominated by a larger requirement of 5 or more oscillators necessary to operate 5 processors, 2 memories, and an I/O port, all of which may be in different LRU's.

Fig. 10 shows the overall failure probability due to lack of equipment for a period of up to 1000 hr. In the short run, the number of buses is critical, while in the long run it is the number of LRU's. The number of power supplies is adequate at all times.

3) *Bus Guardian Unit Failures—Combinatorial Model:* This section discusses the system failure probability due to BGU failures in the enable mode. Although this mode can be made about an order of magnitude less likely than the normal disable failure mode, it is nonetheless present and must be accounted for. As explained earlier, one single BGU may disable a unit from transmitting on a bus, while both BGU's in an LRU must agree before a unit is enabled on a bus. Under the normal circumstances, an active unit (processor, memory, etc.) will be enabled on a single bus. With two BGU's failed in the enable mode, a unit would be enabled on more than one bus. This by itself presents little, if any, problem since three members of a triad transmit in tight synchronism on three buses. However, if the unit enabled on multiple buses fails and does not transmit in synchronism, a number of buses immediately become useless, and this may result in a catastrophic system failure. Thus it takes at least three related failures in a single LRU for the system to fail. The BGU enable mode failures are nonrecoverable. That is, the system can not be reconfigured around a failed BGU. The results for the baseline parameter values are shown in Fig. 10. It is seen that the system failure probability due to this peculiarity of the architecture is at all times insignificant.

4) *Unified Survival Probability Results:* The following conclusions can be drawn from Fig. 10.

1) During a typical commercial flight of one to ten hours the most likely threat of the FTMP computer failure is due to an arrival of two failures so close that system reconfiguration is not possible. The probability of this event, however, is

acceptably low (about  $3 \times 10^{-10}$  per hour) because of high component MTBF's and fast reconfiguration times.

2) There is very little chance that the FTMP computer will run out of spares during a ten-hour flight, assuming that the system initially has all ten LRU's fully operational. In longer flights, however, failure would be quite possible as evidenced by the sharply rising failure probability curve after 50 hours. Lack of equipment is a critical item as far as the dispatch reliability of the computer is concerned, and is discussed in detail in Section IV-C.

3) Finally, the system failure rate due to BGU enable mode failures is substantially lower than other system failure modes. Therefore it does not contribute significantly to the overall system failure probability.

The overall system failure probability due to all causes, up to about 50 hours, is dominated by the probability of failure due to near simultaneous failures. During this time the probability of exhaustion of spares is several orders of magnitude lower. Beyond 100 hours the opposite is true. Strictly speaking, the overall failure probability is a complex function of all the contributing failure probabilities. However, under certain circumstances, it can be approximated very closely by just the predominant failure probability.

#### B. Impact of Intermittent Faults

An intermittent fault in a digital computing system may be defined as a fault that persists only part of the time. Physically, this may correspond to a loose connection between components, a loose bond within a semiconductor device, a temperature sensitive device, etc. Since an intermittent fault manifests itself only a fraction of the time, it injects an additional level of latency to the problem of fault detection. This would lead to longer fault detection and isolation times, thereby reducing the system reliability. The actual extent to which the system reliability would be degraded due to intermittent faults would depend on the degree of latency of the fault. That is, the higher the percentage of time a fault stays in the good state, the higher the chance of it being undetected. With the presence of such a lurking fault in a triad, for example, a second fault in another member of the triad leads to a situation where two out of three members of the triad are at one time or another malfunctioning. If this situation is not redressed promptly by reconfiguration of faulty elements it can result in a catastrophic system failure. On the other hand, the presence of two intermittent faults in two members of a triad can be tolerated as long as one or both of them stay in the lurking mode. This apparently should result in an increased level of fault-tolerance. The following study was undertaken to analyze these contradictory impacts of intermittent faults on the FTMP reliability.

To incorporate intermittent faults in the FTMP survivability models, it is necessary first to define various states and their transition rates corresponding to intermittent faults. In the simplest form, an element with an intermittent fault may be represented by two states: a failed state and a pseudofailed state [18]. In the first state the fault is actually present, that is, use of the element will produce an incorrect output. In the second state, the fault is in a benign mode, and use of the element will not corrupt the output. An intermittent fault will oscillate between these two states with a frequency that is dependent upon the characteristics of the fault. In general, the transition rate from the failed to the pseudofailed state may not be the same as the rate in the other direction (see



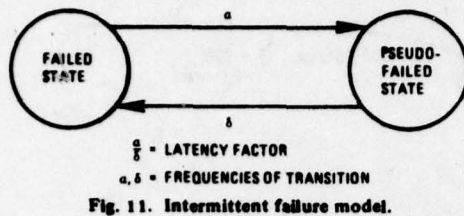


Fig. 11). The ratio of transition rates,  $\alpha/\delta$ , is a measure of the additional latency due to the intermittent nature of the fault. The higher the ratio  $\alpha/\delta$ , the higher is the percent of time a fault stays in the pseudofailed state and is invisible a longer time. For  $\alpha/\delta = 0$ , the intermittent fault really becomes a hard fault since all the time is spent in the failed state.

Certain assumptions have been made regarding the use of this basic model to keep the overall models and the number of parameters tractable. For example,  $\alpha$  and  $\delta$  are assumed to be constant with respect to time. In addition, all faults are assumed to be intermittent with the same transition frequencies and duty cycles. In practice there will be faults with various frequencies which will most likely vary with time as the intermittent faults transition into hard faults. However, the present purpose is to get an insight into how an intermittent fault affects the system survivability. This is best done by simulating a situation where all the failures are intermittent and stay intermittent during the course of investigation.

A Markov process coverage model of a triple modular redundant (TMR) system incorporating the intermittent failure model was developed, as shown in Fig. 12. The reasons for modeling a TMR before going to a full-fledged multiprocessor model are twofold. It involves fewer parameters, making it easier to establish a cause and effect relationship between reliability and various parameters. It also involves fewer states and can be analyzed for a wider range of parameter values. Since the FTMP multiprocessor under investigation is a combination of a number of triads, the TMR results can generally lead to a good understanding of the FTMP reliability behavior.

Fig. 12 shows three different ways in which a catastrophic system failure can result. The first is the occurrence of two simultaneous failures, that is, the failure of a second element before the first failure has been diagnosed and recovered from (transition 2-8). This is the only mode of failure in a TMR system if all the failures were hard failures. However, due to the intermittent nature of our assumed failures, the system can survive even in the presence of two failures as long as at least one of the faulty elements is in the pseudofailed state (states 4, 5, 6, 7). In such a case, the arrival of another failure in the third element (transition 4-8), or the transition of an element from a pseudofailed to a failed state (transition 6-8), leads to a catastrophic system failure. The model was solved numerically for a number of different values of  $\alpha$ ,  $\delta$ ,  $\lambda$ , and  $\mu$ . Some of the important results are shown graphically in Fig. 13. It is found that the failure probability is not a monotonic function of  $\alpha$  or  $\delta$ . However, if the ratio  $\alpha/\delta$  is held constant, the failure probability increases with  $\delta$  as shown in Fig. 13. Similarly, for a constant  $\delta$ , the failure probability generally increases with  $\alpha/\delta$ . In the steady state, the ratio of state probabilities  $P_1$  to  $P_2$  is given by  $\alpha/\delta$ . That is

$$P_1/P_2 = \alpha/\delta.$$

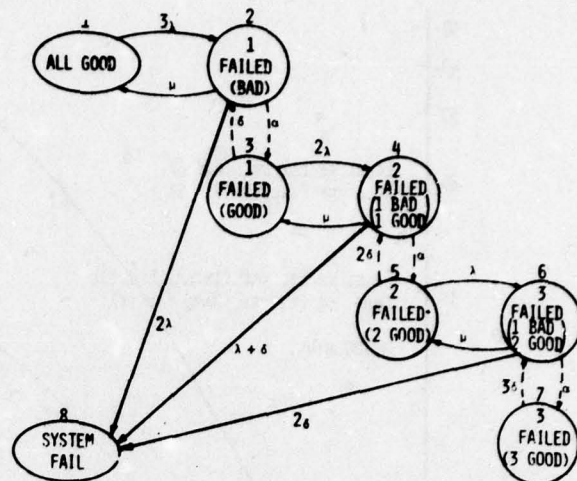


Fig. 12. Intermittent failure model of a TMR-hybrid system.

This is assuming there is no leakage from state 2 to the system fail state 8. Physically, the ratio  $\alpha/\delta$  represents the relative time a fault stays in the lurking mode. That is, the higher the variable  $\alpha/\delta$ , the higher is the latency factor of the intermittent fault. For a fixed ratio  $\alpha/\delta$ , increasing  $\delta$  implies a higher leakage rate from state 4, resulting in a higher failure probability. In other words, since the ratio  $\alpha/\delta$  is fixed, the duty cycle between failed and pseudofailed states is a constant, and, therefore, increasing the frequency of transition between these two states only increases the chance of a lurking fault suddenly crashing the system. It is evident from these results that the worst situation arises where the latency of intermittent faults is high (a high  $\alpha/\delta$ ) and the frequency of transition from pseudofailed to failed state is high (a high  $\delta$ ).

The worst case system failure probability with intermittent faults, for the range of parameters investigated, is about fifty times higher than that due to hard failures (see Fig. 13). The critical frequencies, that is, the worst case  $\alpha$  and  $\delta$ , depend upon the recovery time. The faster the recovery time, the higher these frequencies are. For example, for a recovery time of 36 s, the critical  $\delta$  is  $10^4$  per hour or about 3 Hz, while for a recovery time of a one-quarter second, it is about 30 Hz. Increasing the transition frequencies beyond the critical levels does not further deteriorate the reliability appreciably.

To extend these results to the FTMP computer, a 49-state Markov model was developed. This is basically an expanded version of the 14-state hard failure model described in Section 4.1. All the assumptions of that model carry forward here. This model was solved for the base-line parameter values shown in Table I. The FTMP reliability behavior with respect to  $\alpha$  and  $\delta$  was found to be in close agreement with that of the TMR-hybrid system qualitatively as well as quantitatively. As shown in Fig. 13, the FTMP curve is remarkably close to the TMR curve with typical FTMP failure and recovery rates.

Finally, it should be noted that some of the high-frequency intermittent faults, which could do the most damage, may actually look like hard faults. A fault in a processor module, for example, may cause that module to go out of synchronism with the other two triad members, thereby making its presence felt after it disappears. Therefore, the overall impact of the intermittent faults may not be as severe as suggested here.

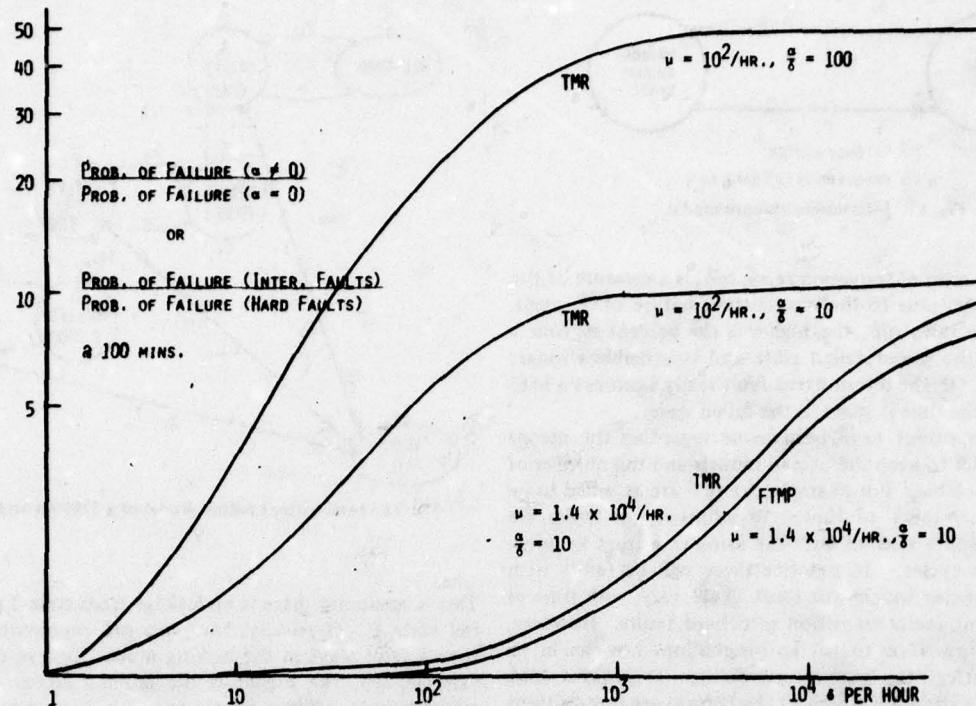


Fig. 13. Influence of intermittency on the system failure rate.

### C. Dispatch Reliability of the FTMP Computer

Availability of equipment, in general, is an important concern in the commercial air transport industry. Availability of avionics equipment, in particular, is economically more important since it tends to be at the heart of "Go/NoGo" decisions. A central computer with digital "fly-by-wire" authority certainly falls into this category. It is imperative, therefore, that the dispatch reliability of the FTMP computer be commensurate with its high survival probability. A preliminary estimate of the dispatch reliability is carried out in this section.

Let the "dispatch minimum complement" (DMC) denote the amount of equipment (processors, memories, etc.) necessary to be operational before take-off for the computer to survive through the flight with a given probability. Using a trial-and-error approach with the combinatorial models of Section IV-A(2), the DMC for the baseline case was found to be as follows:

#### Dispatch Minimum Complement:

Processors	= 8
Memories	= 6
buses	= 4
Power Supplies	= 3

The question to be answered at this point is, how long would it take an initially fully operational FTMP to degrade below the DMC and thereby fail the dispatch criteria? The probability of this event at time  $t$ , assuming no maintenance, is shown as a function of time in Fig. 14. It is seen from this figure that there is a seven per cent chance that the computer will be below the dispatch minimums if the maintenance is scheduled every 300 hours. The probability of requiring unscheduled maintenance can be reduced to just over two per cent by carrying an extra LRU or by shortening the maintenance

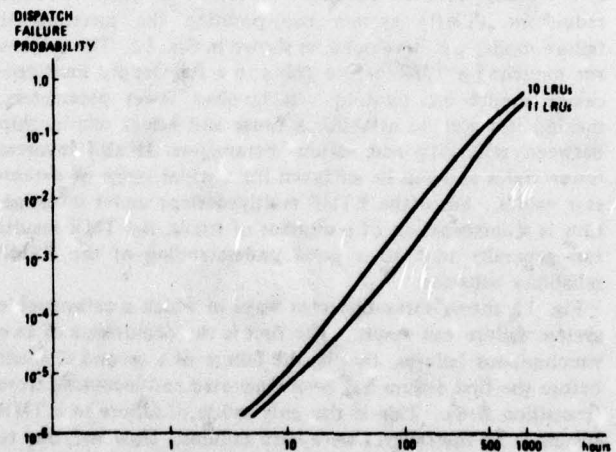


Fig. 14. Dispatch failure probability.

interval to 200 hours. This would seem to satisfy the needs of most airlines as far as the computer dispatch reliability is concerned. Beyond this, however, the dispatch reliability is bounded by the reliability of main power supply units. That is, the dispatch reliability can be improved only by modifying the architecture to include five or more main power supply units.

### V. EXPERIMENTAL RESULTS

In order to demonstrate and validate as many of the design concepts as possible, a breadboard multiprocessor was used to emulate many of the design features of the proposed system. This demonstration was of an integrated nature in that the experimental setup duplicated much of the information en-



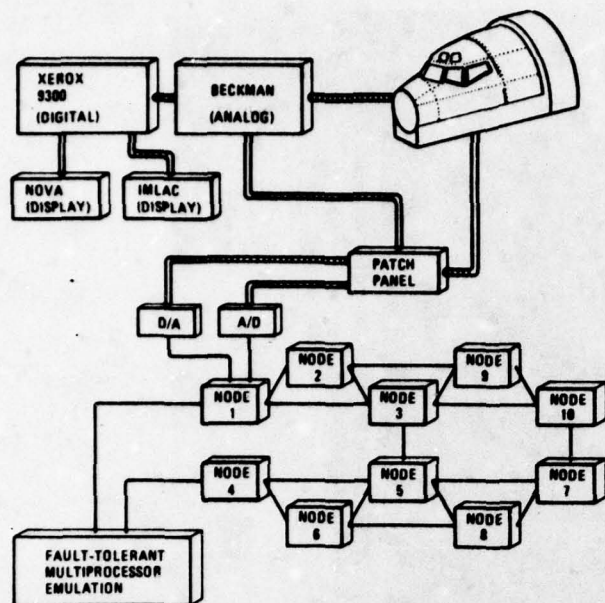


Fig. 15. Experimental simulation system.

vironment which a final product of this nature might encounter, and was therefore able to verify not only the separate design pieces forming the whole, but was also able to confirm predicted interactions between disjoint pieces, and in some cases unearth unexpected interactions.

The basic experimental apparatus consisted of a fault-tolerant multiprocessor, modeled along the lines of the FTMP. The multiprocessor served as the control computer for a Boeing 707 aircraft simulation on a hybrid computer. The experimental fault-tolerant multiprocessor consists of 14 National Semiconductor IMP-16-based processor modules, seven common memory modules of  $2k \times 16$  words, two I/O ports, and ten I/O nodes. The processor modules include  $1k$  RAM/ $1k$  ROM cache memory storage. With the 14 processor modules it is possible to operate up to 4 triads of processors simultaneously. With the seven RAM modules it is possible to operate two memory triads. The redundant data busing system is triply redundant, and each attached module has two Bus Guardian Units associated with it for protecting the bus system. An I/O node remote from the multiprocessor and local to the hybrid computer provides A/D and D/A interfacing to the simulated aircraft as shown in Fig. 15. Fig. 16 is a photograph of the multiprocessor emulation hardware.

#### A. Fault Diagnostic Capabilities

Each processor module of the experimental system includes special circuitry for noting and recording disagreements among the three copies of each bus line. All other modules or receiving elements have only error masking circuits. The error detection circuitry functions as expected. Most faults manifest themselves as bus errors, and are therefore easily detected. Certain classes of latent faults are detected by diagnostic programs which basically force bus errors if a latent fault exists. Records kept by diagnostic programs and fault isolation procedures enable the location of both transient and hard failures.

Most faults are detectable as one of a large class of faults. For example, all processor failures are detected at the bus

without the aid of special diagnostic code to test the processor or knowledge of the fault mechanism. Some special attention to specific failure modes and effects was required to devise latent fault detection programs. While code was not written for unearthing all possible latent faults, sufficient latent testing code was written so as to establish considerable confidence in the method.

The bus isolation mechanism serves as intended and is able to isolate processor failures from the bus system.

This integrated system's demonstration illustrates all significant aspects of the FTMP architecture. It demonstrates the hardware capability to mask faulty unit outputs in the short run, and the capability to detect the fault, isolate the unit, and to reorganize so as to restore system health, all concurrent with normal program activity.

#### B. Software Experience

The software for the demonstration consists principally of executive or system software and applications software. Executive or system software was written and debugged by staff thoroughly familiar with the experimental hardware and design objectives. The applications software was provided by a team which was briefed only in general terms as to the nature of fault recovery mechanisms and the overall system architecture. The applications software team was provided with detailed explanations of the executive-to-applications interfaces and executive services, as well as a reasonably short list of programming constraints.

1) *Multiprocessor Executive*: The multiprocessor executive provides a simple task dispatch mechanism. Tasks awaiting their time of execution are organized in a queue sorted by scheduled start time. As processor triads become free (having finished a previous task) they consult this list and take the next scheduled job. Jobs may be inserted into any relative position of the time queue as long as it remains properly sorted. Executive functions provide for the routine iterative scheduling of the same job step, as might be required for an autopilot iteration, for example. Alternatively, any job, by a call to the executive, can insert a job into the time queue. The executive also handles the removal of a job from the queue when it is taken up for execution.

In addition to the time queue, the executive handles an event queue. Jobs in the event queue have their execution blocked waiting for a particular event to occur. When the event does occur, the affected job is moved from the event queue to the top of the time queue. Jobs can be inserted into the event queue by any job, through a call to the executive. Events can be signaled by the executive or by another job through a call to the executive.

The executive also provides interfaces for all I/O traffic, common memory to/from cache data transfers, real time clock, and for other relatively simple functions commonly thought of as executive-related.

Critical to the success of the demonstration are the executive functions which provide for automatic error logging and recovery. Executive functions perform all common memory to/from cache transfers, and all I/O. During these functions any errors that might occur will become visible. The executive handles the proper logging of the error, schedules recovery action, and, via voting, masks the error for the applications task which was using the executive function. Thus, to the applications task, error handling is completely invisible. Additionally, since hardware monitoring is used, error checking,

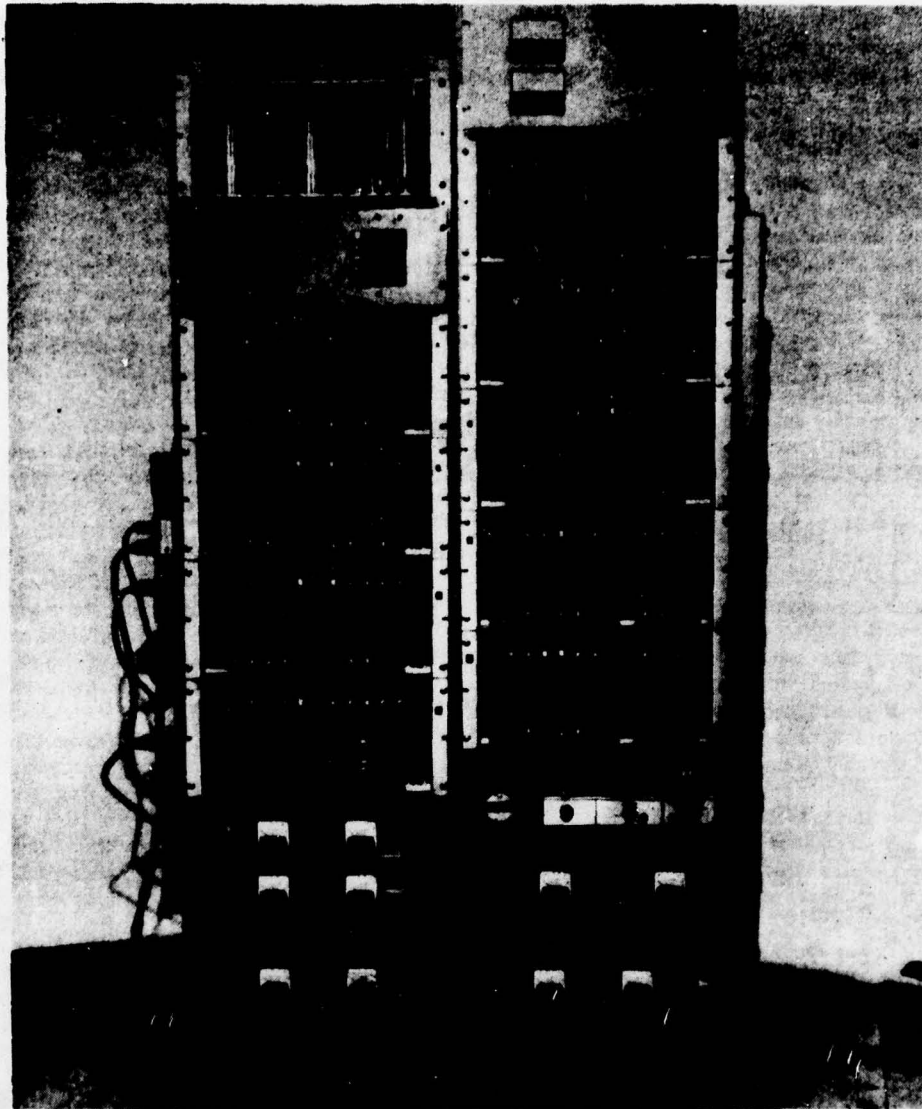


Fig. 16. Multiprocessor emulation hardware.

error masking and majority voting do not impact the applications execution speed.

The executive schedules error diagnostics, latent test routines, and error recovery routines, using basically the same mechanisms used to schedule applications tasks. These executive tasks, running concurrently with the applications tasks, but in different processor triads, maintain the system, repairing faults, searching out latent failures, configuring processor triads and memory triads, and starting and stopping triads as required. Thus in the background, behind the system application, continuous activity is in progress to maintain the integrity of the system, assuring faultless and error-free execution of applications software.

An executive providing these functions was written for the experimental test hardware. Although it is not complete, in that only representative latent faults were tested, the executive does provide the basic facilities for providing error free execution of both executive and applications code. The software framework for latent test procedures is fully developed,

although it is only sparsely populated. Error detection and recovery from all classes of faults is demonstrated in the simulated environment without interfering with the applications tasks.

*2) Cache Memory Management:* The experimental hardware and the proposed future system both have a common memory shared by all processor triads and private cache memories which are part of the processor modules. Programs are executed exclusively out of a processor's cache memory. Clearly, the burden of program loading from common memory, program overlaying, and other functions associated with bringing sections of code from common memory to the cache for execution could not be placed on the applications coding.

In the experimental computer, a software cache-memory management system was provided as part of the executive. At the subroutine call interface, conventions were adopted that provided for the automatic loading of called routines. A last used, first out algorithm clears space in the cache if unused space is not available. If a calling routine is dropped from the



cache to make room for loading of the called routine, it is reloaded by the subroutine return interface.

The efficiency of this process of loading instructions into the cache before execution depends a great deal on the number of times an instruction is executed each time it is brought from common memory. Each word brought from common memory will take about 5  $\mu$ s in the FTMP. Thus one triad executing 190k instructions per second could completely fill the bus capacity. In the experimental system, it is found that the applications programs execute between 10 and 40 instructions for every instruction brought from common memory. If an overall average of 20 can be maintained in the proposed system, a processor triad now projected to have a raw computing power of 200k instructions per second would load the bus with 10k instruction fetches per second. With reasonable allowances made for data transfers and queuing overheads, this suggests a maximum capacity of 4 or 5 processor triads before saturating the memory bus.

## VI. CONCLUSION

### A. Critical Areas of the FTMP Design

The following are areas where the FTMP has required, or will require, special care in conception, analysis, and/or design.

1) The phase-locked redundant clock has presented problems in latent fault exposure and in theoretical validation. Both of these are believed to be solved.

2) Mechanical and electrical design of bus guardians, bus isolation gates, and the buses themselves, must be done with care in order to prevent undesired fault propagation. The engineering prototype design to achieve this is partially complete at this writing.

3) Cold start capability requires the default formation of a triad or the equivalent. This has not yet been designed.

4) Self-test programs must be virtually complete, including perhaps attempts at finding pattern-sensitive failures over a period of time that is large compared to the basic test cycle. These programs will operate by producing bus errors as results of logic malfunctions. They do not need to diagnose the nature of the fault.

5) Mechanisms must be provided in hardware and software to screen or inhibit interferences caused by a lower priority procedure from impinging on a higher priority procedure. The opposite may or may not be possible.

6) Finally, validation must be made effective to a higher degree than ever before. Although some approaches are available, it remains to show how effective they will be.

### B. Summary

The FTMP is a complex multiprocessor computer that employs a form of redundancy related to TMR-Hybrid redundancy, denoted here as Parallel-Hybrid redundancy, in which each major module can substitute for any other module of the same type. Despite the conceptual simplicity of the redundancy form, the implementation has many intricacies owing partly to the low target failure rate, and partly to the difficulty of eliminating single-fault vulnerability.

An extensive analysis of the computer through the use of such modeling techniques as Markov processes and combinatorial mathematics shows that for random hard faults the computer can meet its requirements. It was also shown that the maintenance scheduled at intervals of 200 hr or more can be adequate most of the time. The probability of requiring unscheduled maintenance during this time interval can be

reduced to about two per cent by carrying one or two spare LRU's.

A study of intermittent faults revealed that the longer a fault stays in a pseudofailed state the worse is the system failure probability. Furthermore, high frequency faults also tend to affect the system failure probability adversely. This places an obvious burden upon the computer design and production activities to limit the intermittent failure arrivals and/or their duty cycles and frequencies to values such that the overall failure criterion can be met.

## ACKNOWLEDGMENT

The authors would like to thank Dr. Jean-Claude Laprie of L.A.A.S., Toulouse, France, for his verification of the numerical results for intermittent faults. Dr. John M. Myers and Dr. Anatol Holt were responsible for an analytical validation of the phase-locked fault-tolerant clock.

## REFERENCES

- [1] J. J. Deyst, Jr., and A. L. Hopkins, Jr., "Highly survivable integrated avionics," *Astronautics and Aeronautics*, to be published.
- [2] R. L. Alonso, A. L. Hopkins, and H. A. Thaler, "Design criteria for a spacecraft computer," in *Proc. Seminar on Spaceborne Multiprocessors*, sponsored by NASA Electronic Research Center, Boston, MA, Oct. 1966.
- [3] R. L. Alonso, A. L. Hopkins, Jr., and H. A. Thaler, "A multiprocessor structure," in *Dig. IEEE Computer Conf.*, Chicago, IL, Sept. 1967, IEEE Cat. No. 16CS1.
- [4] A. L. Hopkins, Jr., "A fault-tolerant information processing concept for space vehicles," *IEEE Trans. Comput.*, vol. C-20, no. 11, pp. 1394-1403, Nov. 1971.
- [5] —, "A new standard for information processing systems for manned space flight," *IFAC 3rd Symp. Control Systems in Space*, Toulouse, France, Mar. 1970.
- [6] T. B. Smith, III, "A damage-and-fault-tolerant input/output network," *IEEE Trans. Comput.*, vol. C-24, no. 5, pp. 505-512, May 1975.
- [7] A. L. Hopkins, Jr., and T. B. Smith, III, "The architectural elements of a symmetric fault-tolerant multiprocessor," *IEEE Trans. Comput.*, vol. C-24, no. 5, pp. 498-505, May 1975.
- [8] —, United States Patent No. 4 015 246, Synchronous Fault-Tolerant Multiprocessor System, March 29, 1977.
- [9] J. C. Deckert, M. N. Desai, J. J. Deyst, and A. J. Willsky, "FDDFW sensor failure identification using analytic redundancy," *IEEE Trans. Automat. Contr.*, vol. AC-22, no. 5, pp. 794-803, Oct. 1977.
- [10] A. L. Hopkins, Jr., and T. B. Smith, III, "OSIRIS—A distributed fault-tolerant control system," in *Digest 14th IEEE Computer Society Int. Conf.*, San Francisco, CA, Mar. 1977.
- [11] N. D. Murray, A. L. Hopkins, Jr., and J. H. Wensley, "Highly reliable multiprocessors," in *AGARDograph #224, Integrity in Electronic Flight Control Systems*, F. Kerschels, Ed., AGARD-NATO Neuilly-Sur-Seine, France, Apr. 1977.
- [12] F. P. Mathur, "On reliability modeling and analysis of ultra-reliable fault-tolerant digital systems," *IEEE Trans. Comput.*, vol. C-20, no. 11, pp. 1376-1382, Nov. 1971.
- [13] W. M. Daly, A. L. Hopkins, Jr., and J. F. McKenna, Jr., "A fault-tolerant clocking system," in *Dig. 1973 Int. Symp. Fault-Tolerant Computing*, Palo Alto, CA, June 1973, IEEE Computer Society, IEEE Cat. No. 73CH0772-4C.
- [14] A. A. Avizienis, "Architecture of fault-tolerant computing systems," in *Dig. 1975 Int. Symp. Fault-Tolerant Computing*, Paris, France, June 1975, IEEE Cat. No. 75CH0974-6C.
- [15] A. L. Hopkins, Jr., "Design foundations for survivable integrated on-board computation and control," in *Proc. Joint Automatic Control Conf.*, San Francisco, CA, pp. 232-237, June 1977.
- [16] W. G. Bouricius, W. C. Carter, D. C. Jones, P. R. Schneider, and A. B. Wadia, "Reliability modeling for fault-tolerant computers," *IEEE Trans. Comput.*, vol. C-20, no. 11, pp. 1306-1311, Nov. 1971.
- [17] J-C Laprie, "Reliability and availability of repairable structures," in *Dig. 1975 Int. Symp. Fault-Tolerant Computing*, Paris, France, June 1975, IEEE Cat. No. 75CH0974-6C.
- [18] M. A. Breuer, "Testing for intermittent faults in digital circuits," *IEEE Trans. Comput.*, vol. C-22, no. 3, pp. 241-246, Mar. 1973.
- [19] J. H. Lala and A. L. Hopkins, Jr., "Survival and dispatch probability models for the FTMP computer," in *Dig. 1978 Int. Symp. Fault-Tolerant Computing*, Toulouse, France, June 1978, IEEE Computer Society.

## APPENDIX 6-A

### FUNCTIONAL DESCRIPTION OF THE NAVSTAR GPS X-SET

The Charles Stark Draper Laboratory, Inc. has just completed a study\* that required the functional description of the GPS X-set. Since this description may be useful to some readers, it has been reprinted as an appendix to Section 6.

---

\* Stonestreet, William M., et al, GPS/JTIDS/INS Integration Study Final Report, CSDL Report R-1151, May 1978.



#### 4.1 Functional Description

The purpose of the GPS Baseline Set, which is part of the User Segment of GPS, is to receive the signals transmitted by the GPS satellites and process them to provide highly-precise three-dimensional position and velocity and System Time Information. <sup>(4-1)\*</sup> Each satellite will transmit two distinct Pseudo-Random Noise (PRN) modulated Radio-Frequency (RF) signals at L-band; a Precision (P) navigation signal (10.23 M chips/sec), and a Coarse/Acquisition (C/A) navigation signal (1.023 M chips/sec) at the L1 frequency (1.57542 GHz), and either the P signal or the C/A signal at the L2 frequency (1.2276 GHz).

A functional block diagram of the GPS Baseline Set is shown in Figure 4-1. The set consists of two antennas, two preamplifiers, a receiver, a signal processor (process controller), a data processor, and a power supply. Each of the two antennas receives signals at both the L1 and L2 frequencies. The preamplifiers raise the input signal level thus establishing the input noise figure. The receiver, under control of the signal processor, acquires the satellite signals, tracks the carriers and the codes (either the P or C/A), demodulates the incoming data, and measures the pseudo-range, delta-range and ionospheric propagation delay. The data processor selects the satellites to be tracked and performs the calculations to provide the navigation data.

The GPS Baseline Set has the capability of using an internal-reference oscillator or an external-reference oscillator as a frequency source and/or an external clock for accurate time-of-week information. This set is also capable of using data from an Inertial Measurement Unit (IMU) to provide improved velocity and position estimates. The IMU data is used in the navigation filter.

The GPS Baseline Set is essentially the X-Set designed by the Magnavox Government and Industrial Electronics Company, Advanced Products Division, Torrance, CA.

Detailed descriptions of each part of the GPS Baseline Set are given in the following sections taken from Reference 4-1.

##### 4.1.1 Antennas and Preamplifiers

Two antennas are used to form a quasi-omnidirectional antenna. Both antennas receive L1 (1575.42 MHz) and L2 (1227.6 MHz). Normally one antenna is selected on the basis of its beam pattern to track satel-

\* Superscript numerals refer to similarly numbered references in the List of References at the end of this appendix.

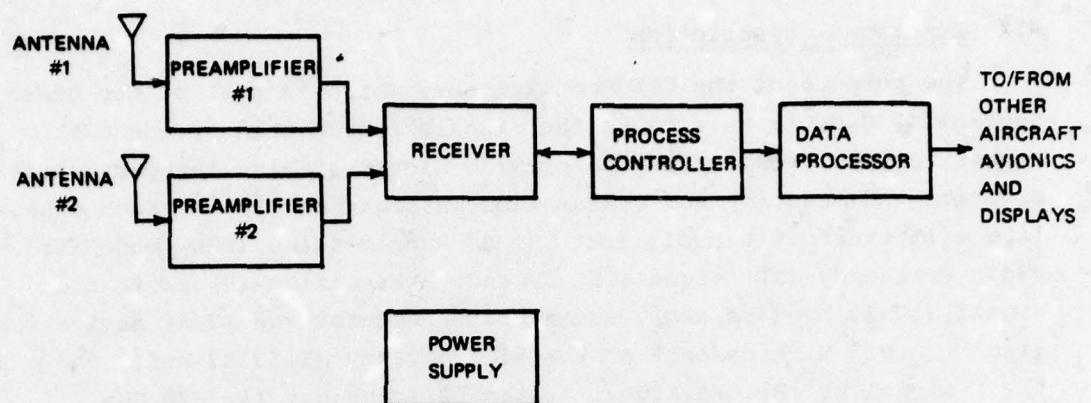


Figure 4-1. Functional block diagram of GPS baseline set.

lites with low elevation angles, while the other tracks satellites with higher elevation angles. (Due to its lower gain at low elevation angles the latter antenna provides higher anti-jamming capability against ground jammers.)

A preamplifier is located at each antenna. As an input, the preamplifier accepts either the antenna signal or a calibration signal provided by the receiver. A block diagram of the preamplifiers is shown in Figure 4-2. The directional coupler connects either the antenna or calibration signals to the diplexer. The diplexer isolates the L1 and L2 signals from each other and from other interfering signals such as phase-arrayed radar signals. The L1 and L2 signals are then amplified and summed together for transfer to the RF converter at the receiver. Isolation and amplification of the L1 and L2 signals in this manner prevents these signals from jamming each other. Table 4-1 presents the preamplifier performance characteristics.

#### 4.1.2 Receiver

The receiver portion of the GPS Baseline Set consists of an RF converter, IF signal switches, frequency synthesizer, reference oscillator, code channel, four carrier channels and power supply. A functional block diagram of the receiver is given in Figure 4-3. The RF converter down-converts the incoming radio frequency signals, at the L1 and L2 frequencies, to intermediate frequencies, amplifies these signals and switches then to the four carrier channels and the code channel. The frequency synthesizer generates all of the stable LOs used by the receiver. The reference oscillator provides the basic 5.115 MHz reference frequency to the frequency synthesizer and receiver timing



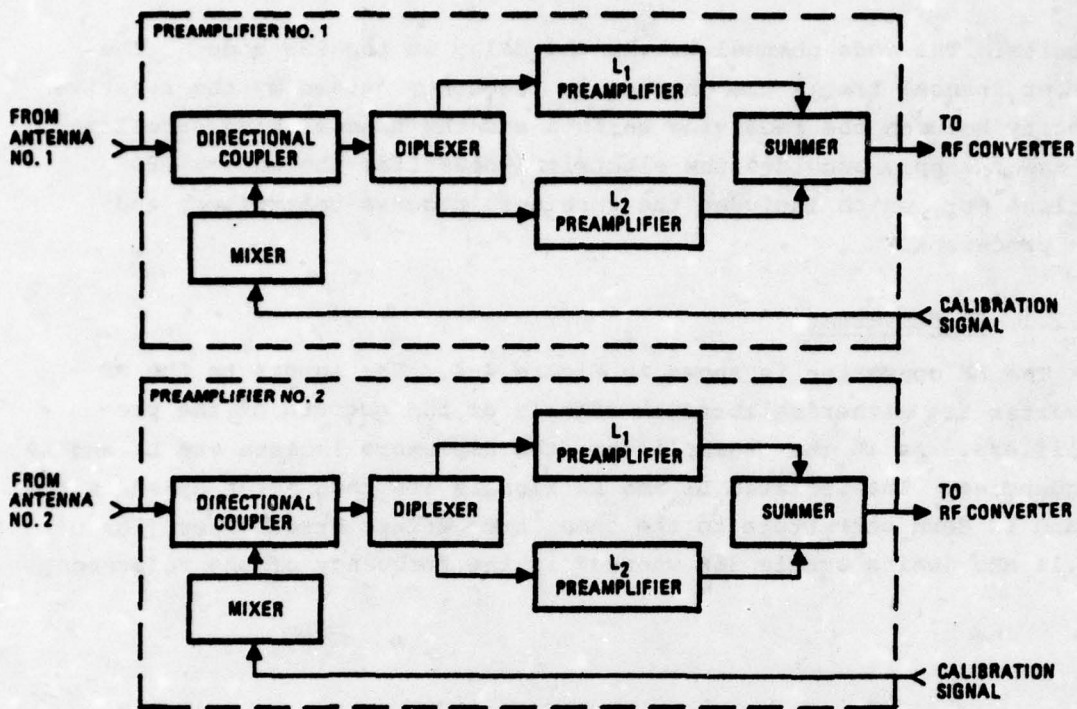


Figure 4-2. Preamplifiers.

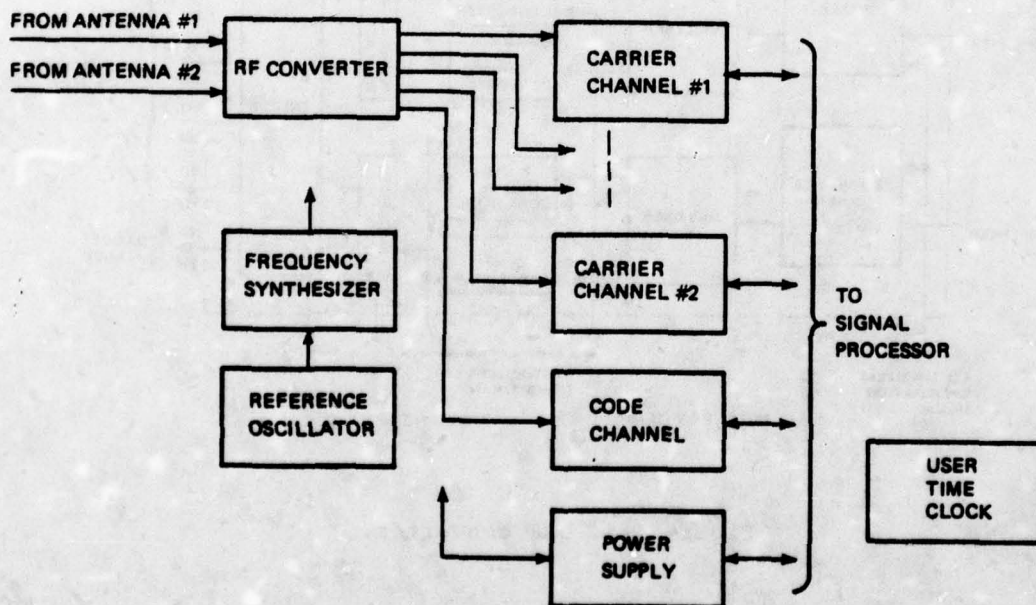


Figure 4-3. Functional block diagram - GPS baseline receiver.

circuits. The code channel tracks the delay in the PRN code. The carrier channel tracks the changes in frequency caused by the relative velocity between the receiving antenna and the transmitting satellite. The power supply provides the electrical power for the entire GPS Baseline Set, which includes the receiver, process controller, and data processor.

#### 4.1.2.1 RF Converter

The RF converter is shown in Figure 4-4. The inputs to the RF converter are either calibration signals or the outputs of the preamplifiers. As in the preamplifier, the diplexers isolate the L1 and L2 frequencies. The isolated L1 and L2 signals are then heterodyned in the L1 and L2 down converters to the same Intermediate Frequencies (IF) of 184.14 MHz (which equals  $36F$  where  $F$  is the frequency of the reference

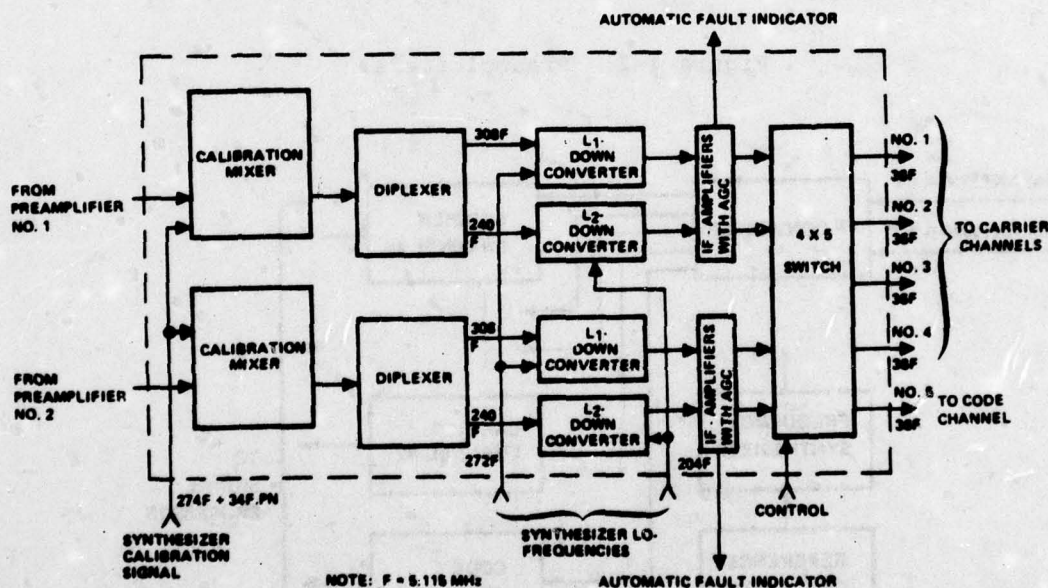


Figure 4-4. RF converter.



Table 4-1. Preamplifier performance specifications.

Description	Characteristics
No. of antenna signal inputs	1
No. of calibration signal inputs	1
No. of RF signal outputs	1
Signal waveform $L_1/L_2$	FDM
Nominal input/output center frequencies ( $F_o$ ):	
$L_1$	1575.42 MHz
$L_2$	1227.60 MHz
$L_1$ and $L_2$ bandwidth selectivity:	
At -1.0 dB	$\pm 9$ MHz
At -3.0 dB	$\pm 12$ min. $\pm 17$ max. MHz
At -70.0 dB	$\pm 70$ MHz
Nominal input/output impedances	50 ohm
Max. input/output vswr ( $F_o \pm 8$ MHz)	1.5:1
Max. noise figure	3.5 dB
Reference preamplifier input:	
Remote located	100 ft. max.
Cable loss	4 dB max.
Input signal levels (including J/S):	
Max.	-50 dBW
Min.	-180 dBW
Dynamic range (noise level to 1-dB compression)	130 dB
Burnout protection	0 dBW min.
Gain at $F_o$	30-34 dB
Phase linearity ( $\pm 8.0$ MHz)	$\pm 5$ deg
Reverse isolation (min.)	30 dB
Decoupling for calibration signal injection	20 to 30 dB
Calibration signal input level:	
Max.	-120 dBW
Min.	-140 dBW
Group delay variation (over $\pm 8.0$ MHz range)	10 nsec
Isolation ( $L_1$ to $L_2$ )	50 dB min.
Calibration signal input	274F $\pm$ 34 F·PN
Input signal level	-34 dBW $\pm$ TBD
Output signal level	-37 dBW $\pm$ TBD
Input/output impedance	50 ohm
	Note: $F = 5.115$ MHz

oscillator, 5.115 MHz) and then amplified in the IF amplifiers. The IF amplifiers utilize a total-power noncoherent Automatic Gain Control (AGC) circuit followed by clippers which clip at an output level approximately 3 dB above the noise level. The outputs of the four IF amplifiers go to a 4 x 5 switch which can switch any one of the four inputs (two antennas, two frequencies each) to any one of the five outputs (four carrier channels, one code channel). The RF-converter performance characteristics are presented in Table 4-2.

#### 4.1.2.2 Frequency Synthesizer/Reference Oscillator

The frequency synthesizer (shown in Figure 4-5) generates all of the stable continuous wave signals used by the receiver for timing and as local oscillators. There are five functional blocks in the frequency synthesizer: internal reference oscillator (5.115 MHz), reference converter (Figure 4-6), low frequency synthesizer (Figure 4-7), L-band synthesizer (Figure 4-8), and calibration signal synthesizer (Figure 4-9). If an external oscillator is present, the synthesizer detects its presence and phase locks the 5.115 MHz reference oscillator to the 5 MHz external oscillator. The reference oscillator can be adjusted over a 4 Hz range. The 10.000 to 5.115 MHz phase-locked loop is depicted in Figure 4-6. This phase-locked loop has a bandwidth of 1 Hz. Tables 4-3 and 4-4 present the performance characteristics of the reference oscillator and frequency synthesizer, respectively.

#### 4.1.2.3 Carrier Channel

The carrier channel shown in Figure 4-10 consists of a local reference generator/correlator, signal conditioner, carrier rate multiplier/incremental-phase modulator, code-rate multiplier/incremental-phase modulator, code generator, and address selector/data director. The functional operation of each of these units is described in the following paragraphs.

The local-reference generator/correlator (Figure 4-11) heterodynes the carrier estimate (nominally  $0.25F = 1278.75$  kHz) from the carrier rate multiplier/incremental phase modulator to a nominal frequency of  $29.25F = 149.61375$  MHz. The estimated code (in this case



Table 4-2. RF converter performance specifications.

Description	Characteristics
No. of RF inputs	2
No. of LO inputs	2
No. of IF outputs	5
Nominal RF input center frequencies ( $F_0$ ):	
$L_1$	1575.42 MHz
$L_2$	1227.60 MHz
Nominal IF output center frequency	184.14 MHz
$L_1/L_2$ -bandwidth selectivity:	
At -1 dB	$\pm 9$ MHz
At -3 dB	$\pm 11$ min., $\pm 17$ max. MHz
At -70 dB	$\pm 150$ MHz
Nominal input/output impedances	50 ohm
Max. input/output VSWR	1.5:1
Max. noise figure	23 dB
Input signal levels:	
Max.	-50 dBW
Min.	-150 dBW
Dynamic range (gain compression to 1 dB)	100 dB
Pulse-clipping level (output referenced):	-40 dBW
Overload recovery	100 nsec max.
Gain at $F_0$	55 dB
Output power level at 1 dB gain compression	-45 dBW
Phase linearity ( $\pm 8.0$ MHz)	10 deg
Output IF switching time:	2 $\mu$ sec max.
Isolation:	
Between down-conversion channels	30 dB
Between LO inputs	20 dB
Between IF outputs	30 dB
Between LO inputs and IF outputs	30 dB
Calibration signal:	$274F \pm 34F \cdot PN$
Output signal level	-120 dBW $\pm 5$
Input signal level	-50 dBW $\pm 5$
Input/output impedance	50 ohm
Calibration command:	
Signal levels	TTL
	Note: $F = 5.115$ MHz

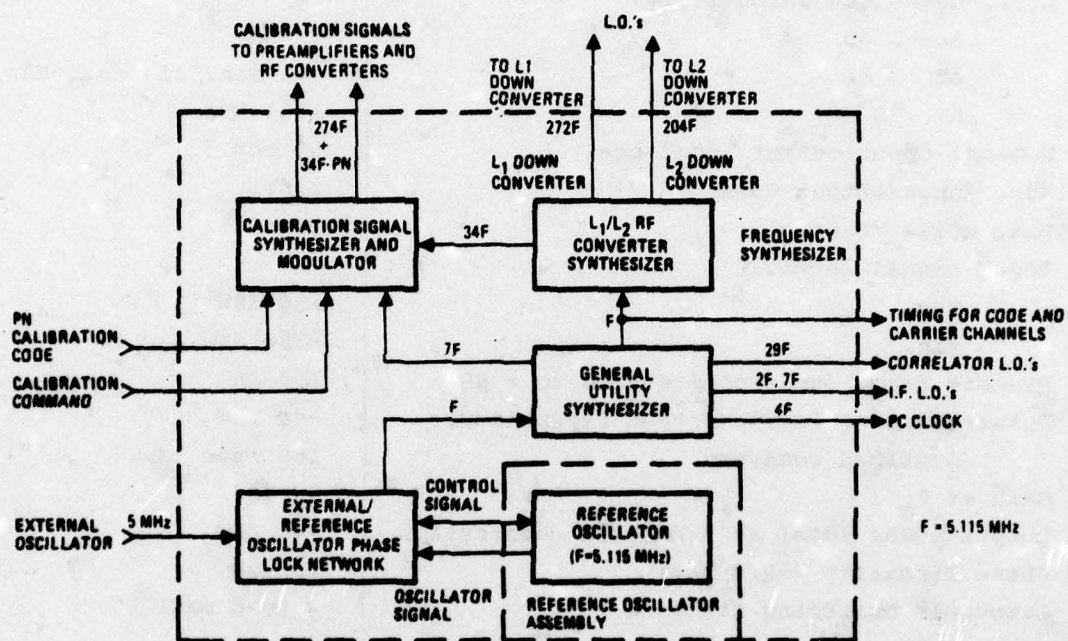


Figure 4-5. Frequency synthesizer/reference oscillator.





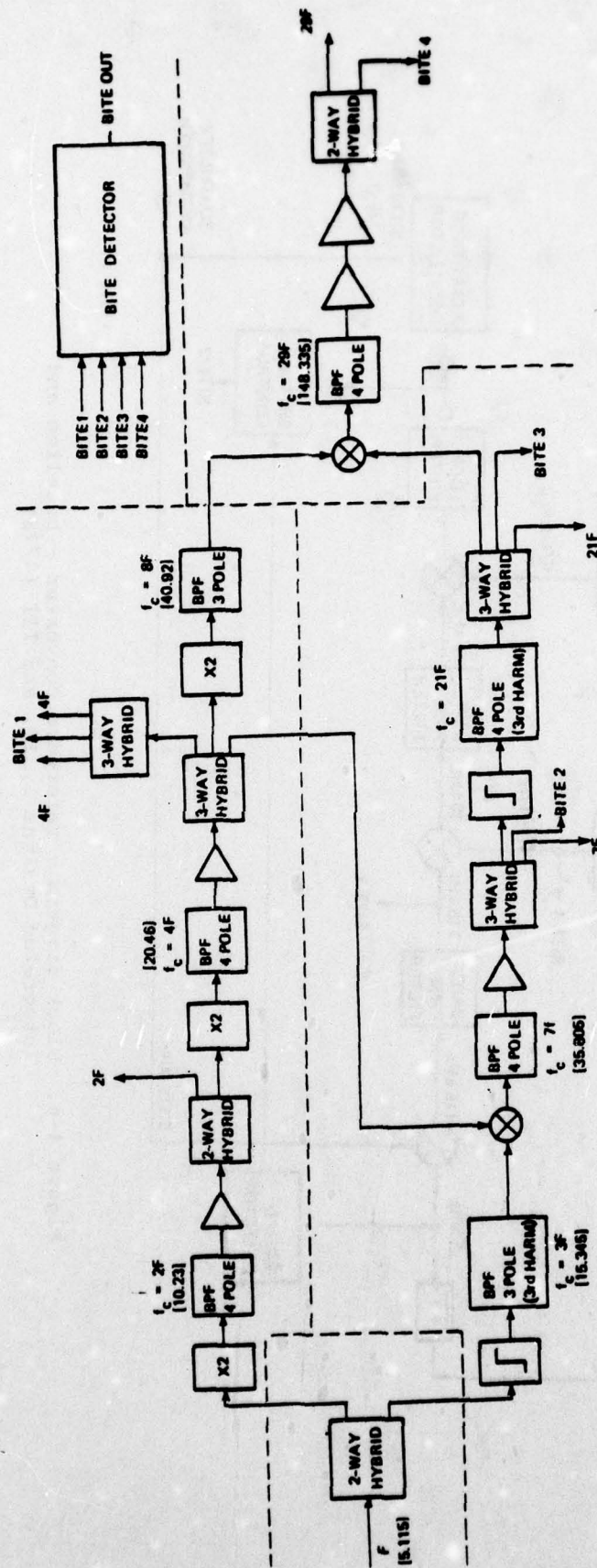


Figure 4-7. Block diagram - low-frequency synthesizer - baseline and Integrated Designs I and II.



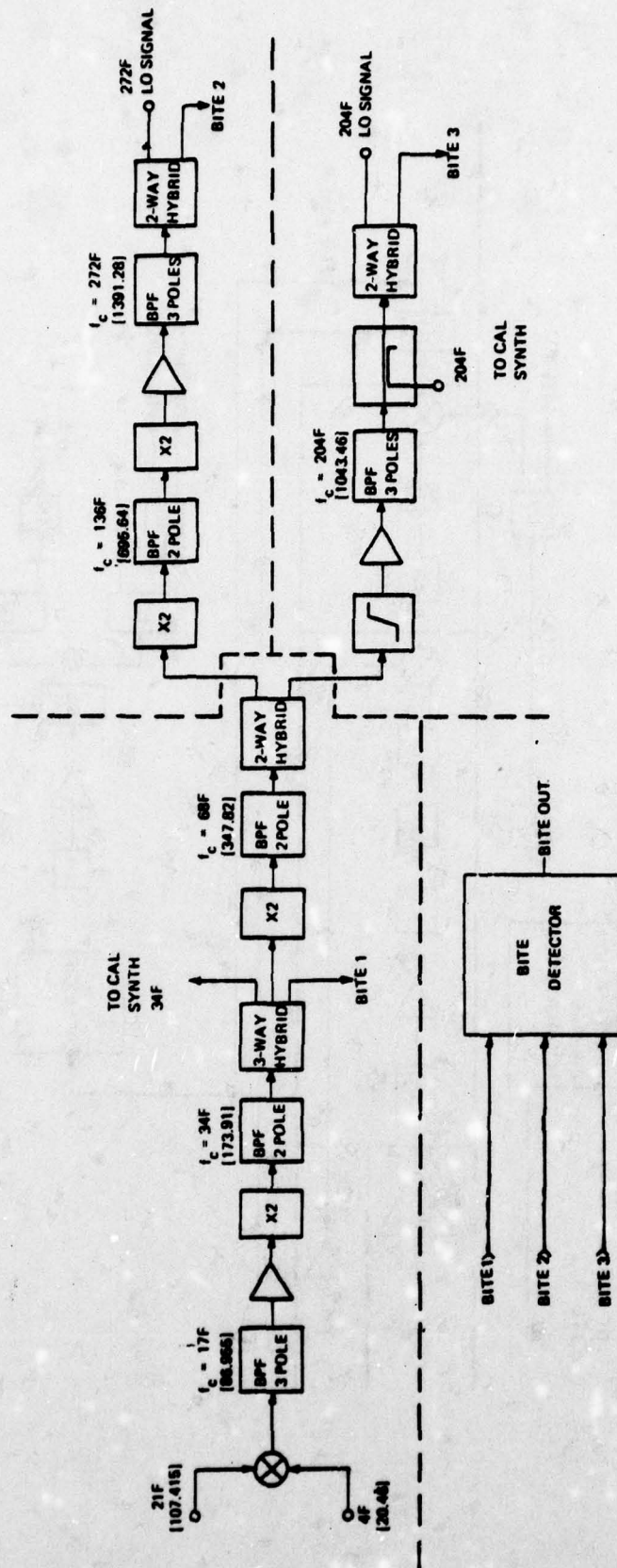


Figure 4-8. Block diagram - L-band synthesizer - baseline and Integrated Designs I and II.

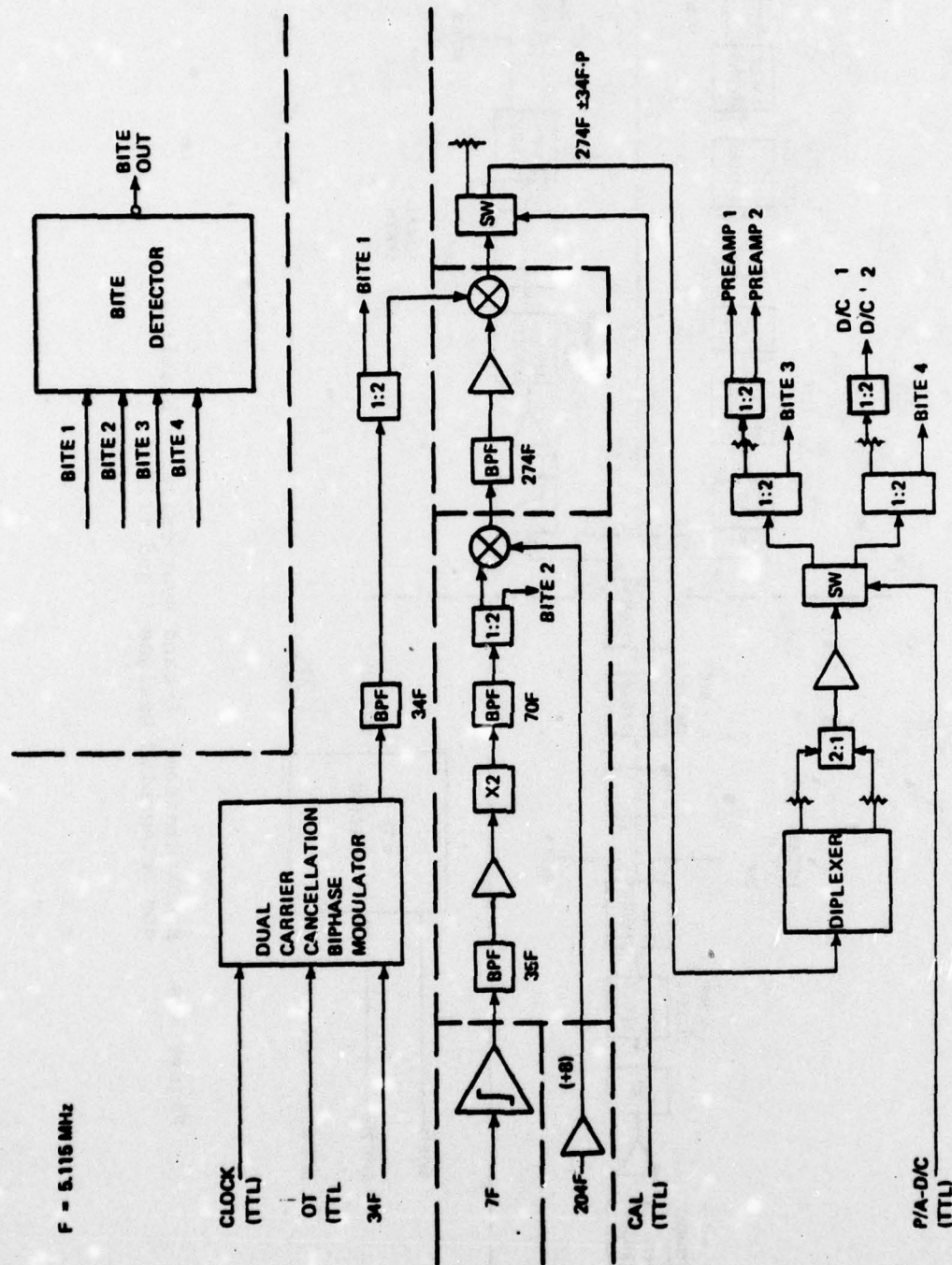


Figure 4-9. Calibration signal synthesizer.



Table 4-3. Reference oscillator performance specifications.

Description	Characteristics
Nominal output frequency	5.115 MHz
Frequency adjust range:	
Coarse	$\pm 10$ Hz min.
Fine	$\pm 1$ Hz min.
Output level (rms)	0.5 V min.
Output load	50 ohm (nominal)
Temperature range:	
Operation	-20 to +70°C
Storage	-65 to +125°C
$\frac{\Delta F}{F}$ Stability:	
Total frequency deviation over entire temperature range	$< 1 \times 10^{-9}$
Short term	$< 1 \times 10^{-10}$ /sec
Aging rate	$< 1 \times 10^{-9}$ /24 hr
Voltage	$< \pm 1 \times 10^{-9}$ /±5 percent
Loading	$< \pm 1 \times 10^{-9}$ /10 percent
Vibration	$< 2 \times 10^{-9}$ /g
Shock	$< 2 \times 10^{-9}$ /g
Acceleration	$< 3 \times 10^{-9}$ /g
Stabilization:	
From temperature:	-20°C to +70°C
5 Minutes	$< 1 \times 10^{-7}$
30 Minutes	$< 2 \times 10^{-9}$
Frequency pulling range	0.4 ppm

Table 4-4. Frequency synthesizer performance specifications.

Description	Characteristics
Receiver reference oscillator frequency (F)	5.115 MHz
Synthesized frequencies	.2F, .4F, 7F 17F, 21F, 29F, 34F, 204F, 272F, 274F
Power level for synthesized frequencies	-23 ± 3 dBW
Phase-noise contribution of synthesizer:	
LO frequencies (rms)	2 deg
Timing signals	2 deg
Calibration signals (rms)	10 deg
Spurious level:	
LO outputs	-50 dB
Timing signals	-40 dB
Calibration signal	-30 dB
External input reference oscillator frequency:	
Frequency	5.0 MHz
Signal level (rms)	1.0 V
Nominal input/output impedances	50 ohm
Max. VSWR	2:1 max.
Isolation:	
Between LO outputs	50 dB
Between LO and calibration signal outputs	50 dB
Between all outputs and reference oscillator input	50 dB
Between all outputs and code signal input	40 dB
Calibration signal	274F + 34F · PN Note: F = 5.115 MHz



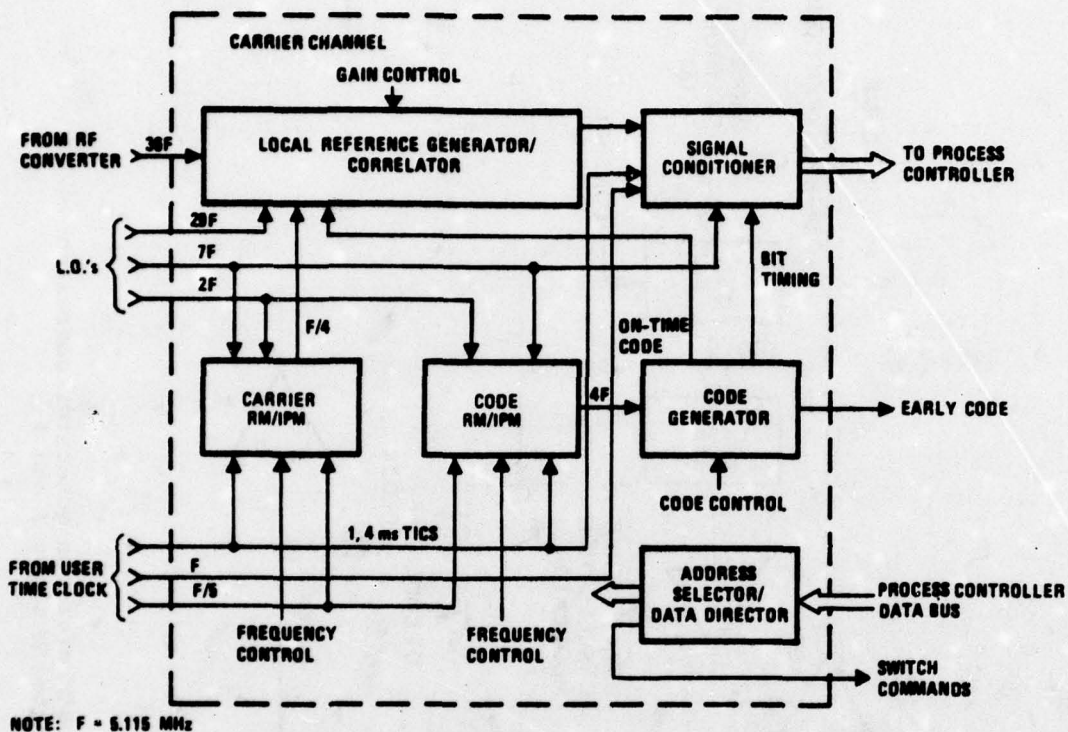
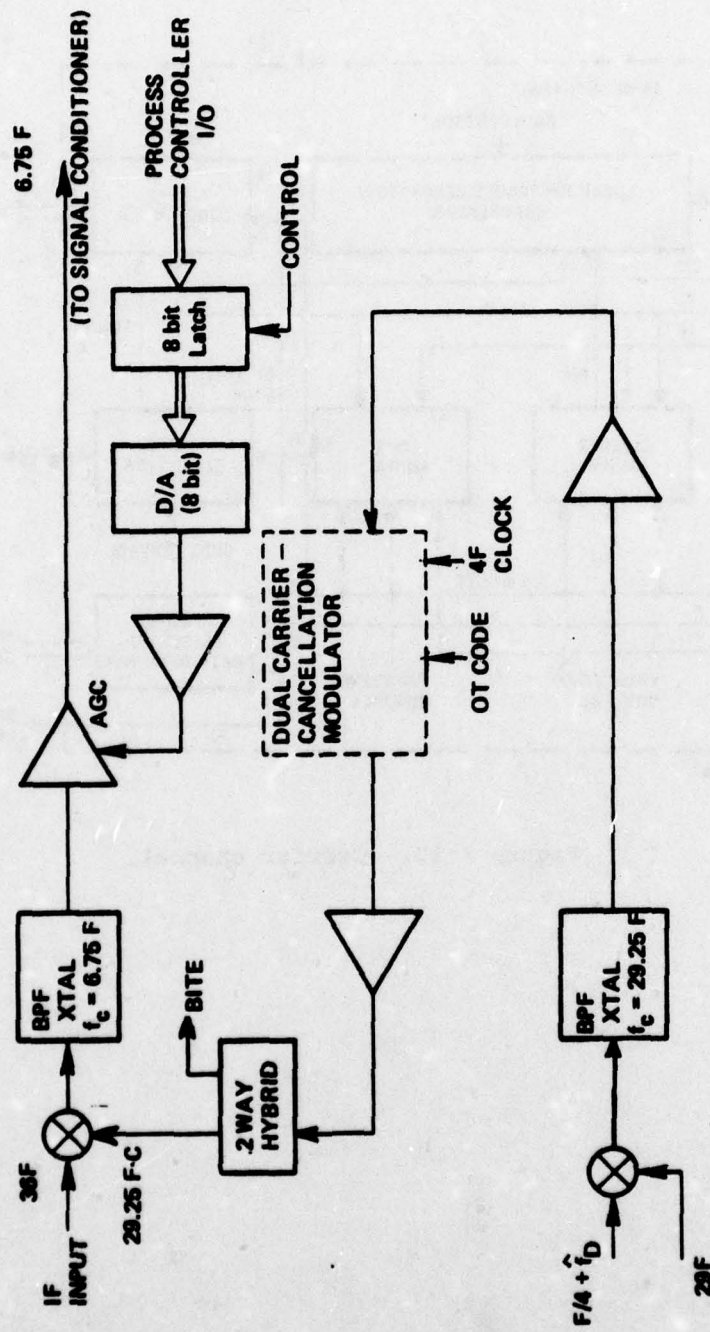


Figure 4-10. Carrier channel.



$F = 5.115 \text{ MHz}$

Figure 4-11. Block diagram - local reference generator/correlator - baseline and Integrated Designs I and II.



the on-time code estimate) is superimposed upon this signal to form the local reference for this channel. The local reference generator/correlator also amplifies the signal from the RF converter in a coherent AGC (controlled by the process controller) and correlates it with the local reference for this channel. This correlation generates a second IF of nominally  $6.75F = 34.52625$  MHz which goes to the signal conditioner.

The signal conditioner (Figure 4-12) heterodynes the output of the local-reference generator/correlator to the detection frequency, nominally  $0.25F = 1278.75$  kHz. It then correlates this signal with quadrature signals of fixed frequency  $0.25F$  and integrates the outputs for a period of time (T). At the end of this time interval the outputs of the integrators are sampled and reset. The samples are converted to eight-bit binary words. Depending upon the operation the receiver is performing, T is either one or four milliseconds. In general, if the receiver is in an acquisition mode, T is one millisecond, otherwise T is four milliseconds.

The carrier Rate Multiplier/Incremental-Phase Modulator (RM/IPM) is essentially a digital Voltage-Controlled Oscillator (VCO). Figure 4-13 is a block diagram of the carrier RM/IPM. Every 4 ms the process controller supplies a twelve-bit control word, FREQ, to the RM and a one-bit control word to the IPM. The output frequency of the RM is equal to

$$(F/5) \frac{FREQ}{4092}$$

where  $F = 5.115$  MHz. The IPM operates in the following manner. The phase of the output signal is advanced or delayed by dividing a reference signal of frequency  $2F$  by either three, four, or five. The divider is controlled by the output of the RM and the carrier sign bit. The output of the RM indicates whether the output phase should be changed or not and the carrier sign bit indicates in which direction the phase should change to drive the loop error to zero. If the phase is to remain the same, the divider divides by four. If not, the divider divides by either three or five depending upon whether the phase is to be advanced or delayed. Then the IPM divides the output of the variable-modulus divider by two and heterodynes the signal with the fixed frequency  $1.75F = 8.95125$  MHz. It selects the sum-frequency,  $2F = 10.23$  MHz, output of the heterodyner with a three-pole band-pass filter and divides

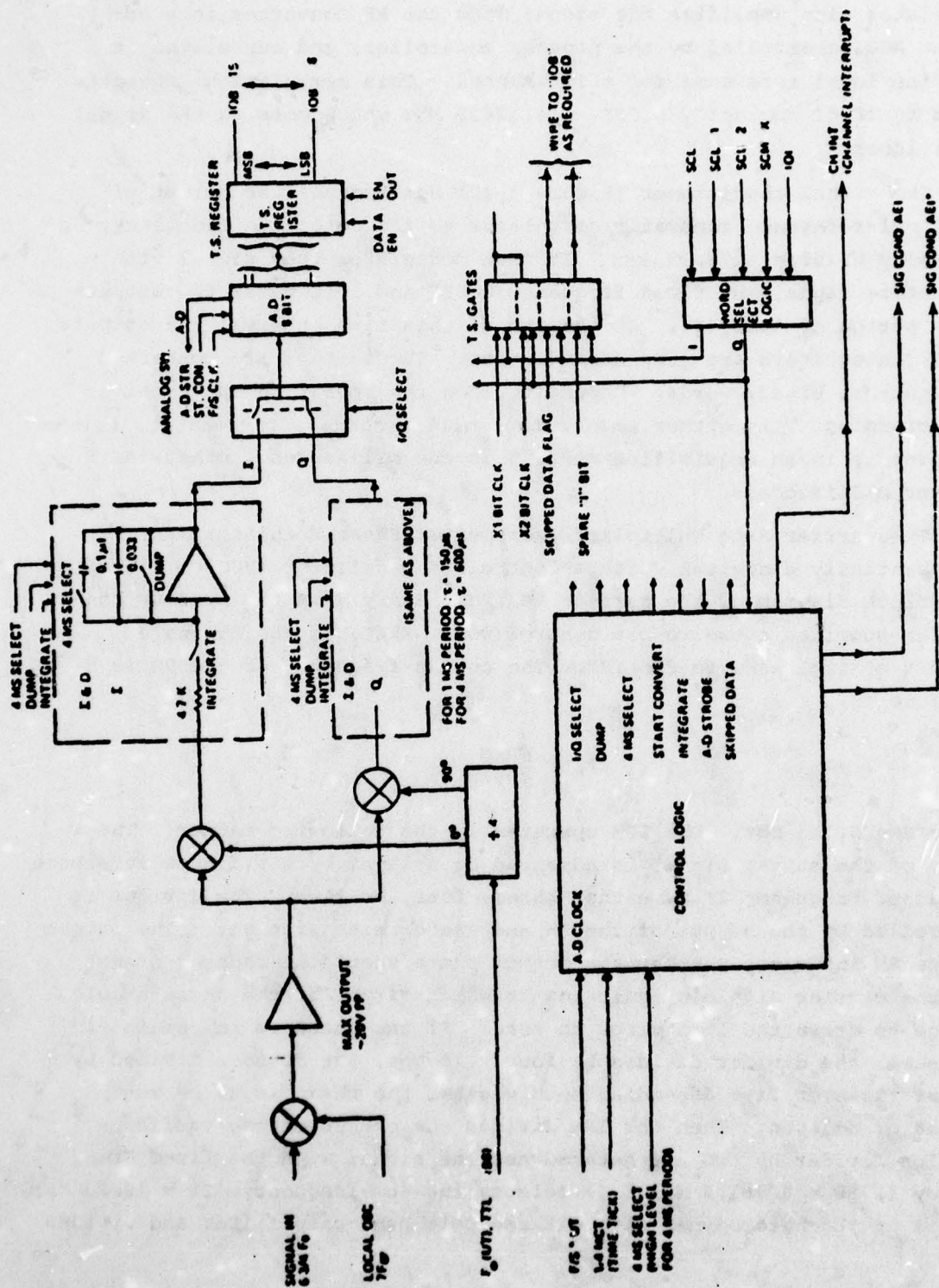


Figure 4-12. Block diagram - signal conditioner baseline and Integrated Designs I and II.



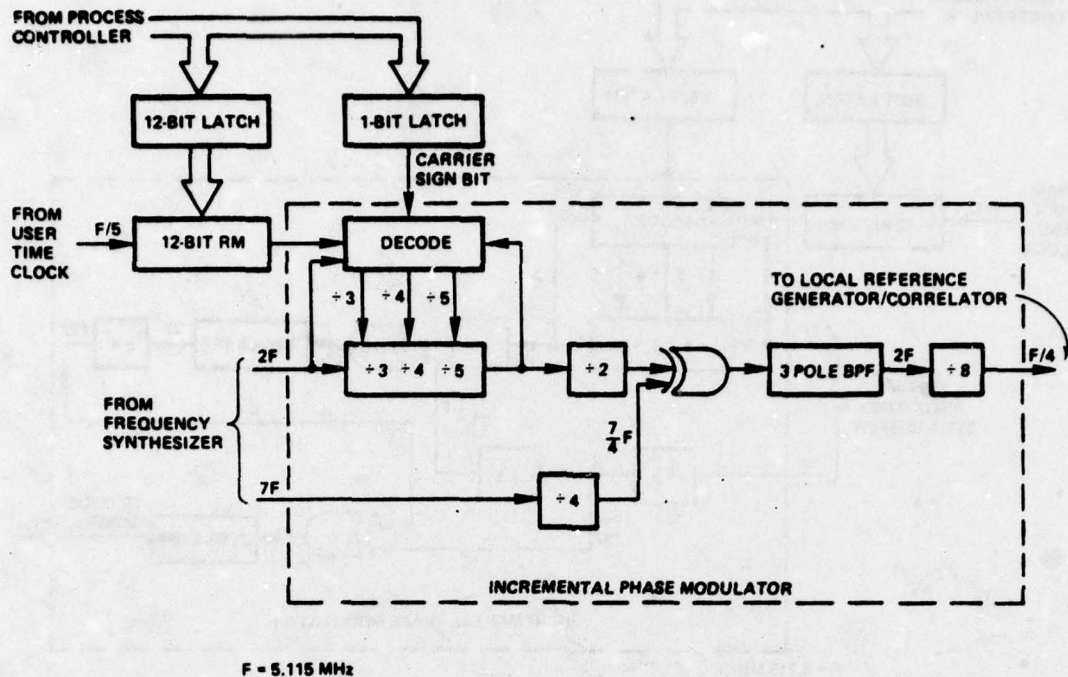


Figure 4-13. Carrier rate multiplier/incremental phase modulator.

this signal by eight to produce the phase-modulated output signal of frequency  $0.25F = 1278.75 \text{ kHz}$ . A change of one least-significant bit in **FREQ** causes a change of  $1/64$  of a cycle every 4 ms in the output of the **RM/IPM**.

The code **RM/IPM** is shown in block diagram form in Figure 4-14. Its operation is similar to that of the carrier **RM/IPM** except that only a five-bit word is used to control the **RM** and that the output of the **IPM** is generated by dividing the output of the first band-pass filter by two, heterodyning this signal with  $3.5F = 17.9025 \text{ MHz}$ , and selecting the sum-frequency,  $4F = 20.46 \text{ MHz}$ , output of the heterodyner with a two-pole band-pass filter. As with the carrier **RM/IPM** a change of one least-significant bit in the control word causes a change of  $1/64$  of a chip every 4 ms in the output of the code **RM/IPM**.

The code generator shown in Figure 4-15 generates both the **C/A** (Gold) and **P** codes. It also generates channel interrupts at either 1- or 4-ms periods and provides a bit clock. Four twelve-stage linear feedback shift registers are used to generate the **P** code. The outputs of the **X1A** and **X1B** registers are modulo-2 summed to form the output of the **X1** register. The outputs of the **X2A** and **X2B** registers are added

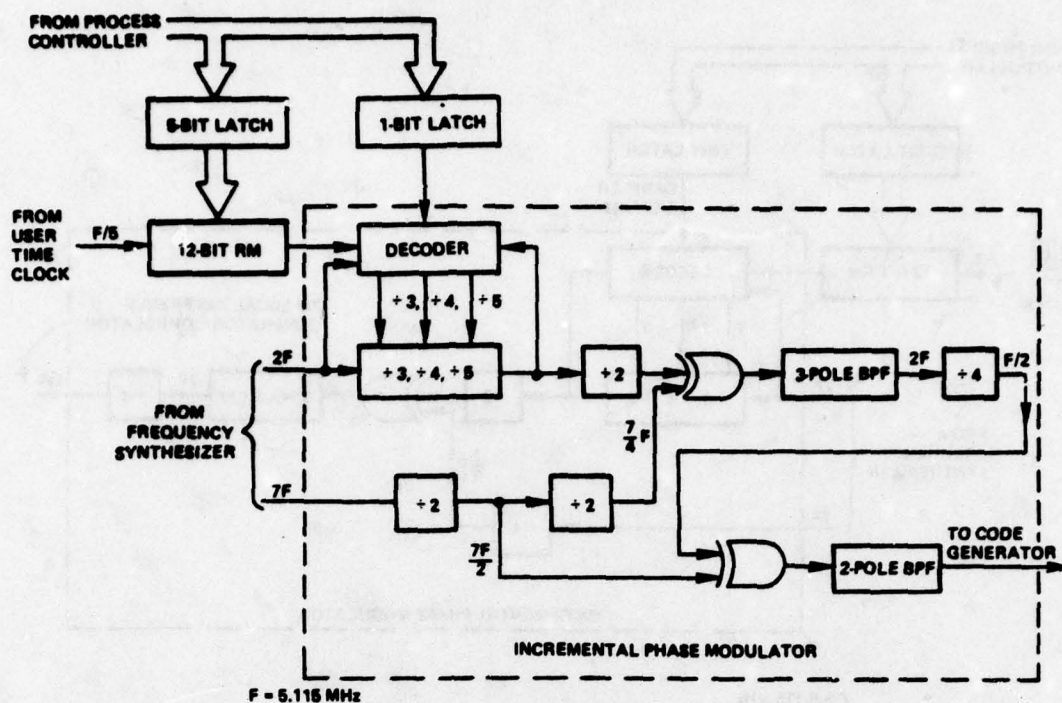


Figure 4-14. Code rate multiplier/incremental phase modulator.

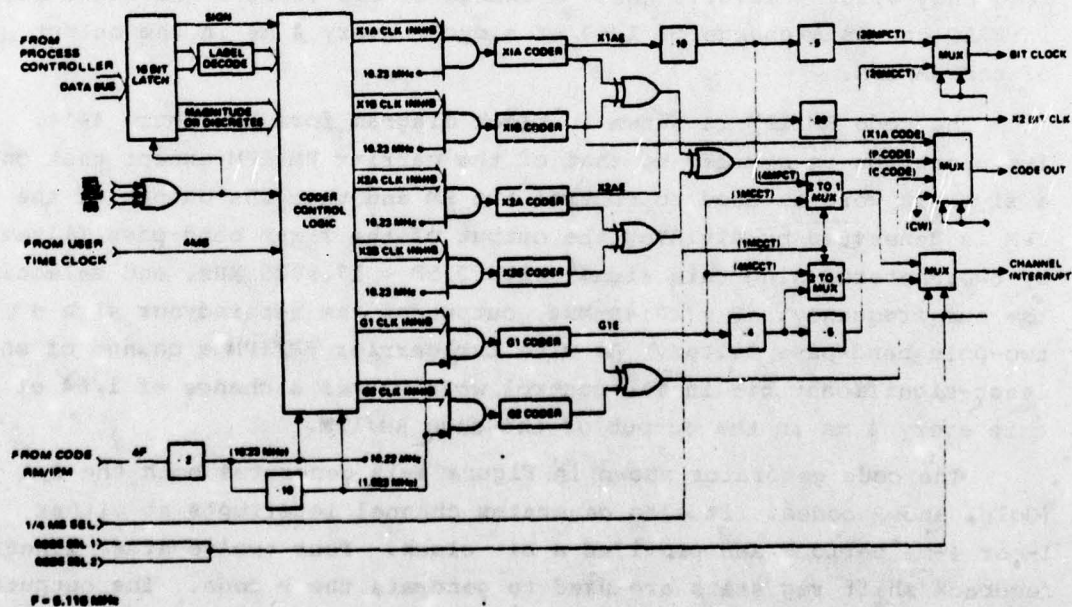


Figure 4-15. Code generator.



modulo-2 to generate the output of the X2 register. The P code is formed by modulo-2 adding the outputs of the X1 and X2 registers. Upon receiving the proper commands from the process controller the code generator can set the epoch of the X1 register and slew the X2 register to any value in less than 1.62 seconds. The C/A (Gold) code is generated by modulo-2 summing the outputs of two ten-stage linear feedback shift registers. The input to the P and C/A code generators is the output of the code RM/IPM divided by two and twenty, respectively. Both the P and C/A codes are generated four chips early and then delayed by 3.5, 4, and 4.5 chips for the early, on-time, and late correlations, respectively.

The address selector/data director in Figure 4-10 selects the data that is intended for that channel from the data bus and directs it to the proper device, e.g., carrier or code RM/IPMs.

#### 4.1.2.4 Code Channel

Whereas there are four carrier channels (one for each satellite signal being tracked), there is only one code channel which is time shared between each of the signals being received. The sequence of code channel measurements is shown in Figure 4-16. First the code error for channel one is measured. This requires a time interval corresponding to the two data bits of equivalently forty milliseconds. Then the code errors for channels two through four are measured. Next the channel-one L2 measurement is made. There is a ten-millisecond guard

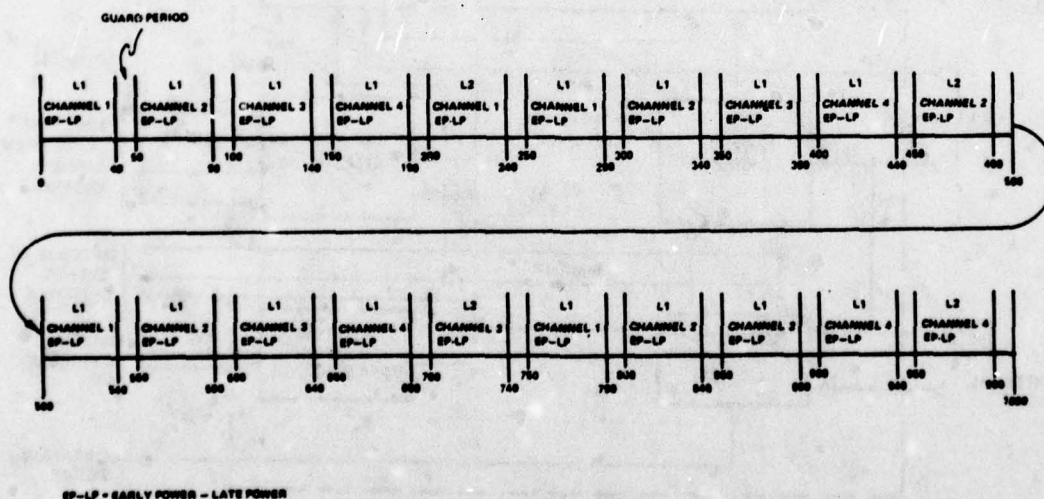


Figure 4-16. Time sharing sequence of the code channel while in the tracking mode.

band between measurements. This 250-millisecond cycle is repeated with each fifth measurement being an L2 measurement for a different channel. Thus the update rates for code-error measurements and L2 measurements are 250 and 1000 milliseconds, respectively.

A block diagram of the code channel is shown in Figure 4-17. It consists of a local reference generator/correlator (Figure 4-11), signal conditioner (Figure 4-12), carrier RM/IPM (Figure 4-13), programmable digital delay (Figure 4-18), user-time clock (Figure 4-19), and an address selector/data director. The operations performed by the local reference generator/correlator, signal conditioner, address selector/data director and carrier RM/IPM are the same as the operations performed by the local reference generators/correlators, signal conditioners, address selectors/data directors, and carrier RM/IPMs of the carrier channels except that in the local reference generator/correlator the incoming signal is alternately correlated with the early and late codes instead of the on-time code. (It should be noted that the code loop requires a separate carrier RM/IPM because of the manner in which L2 measurements are made. If this were not the case, the outputs of the carrier RM/IPMs on the carrier channels could simply be routed to the code channel and properly switched for correlation)

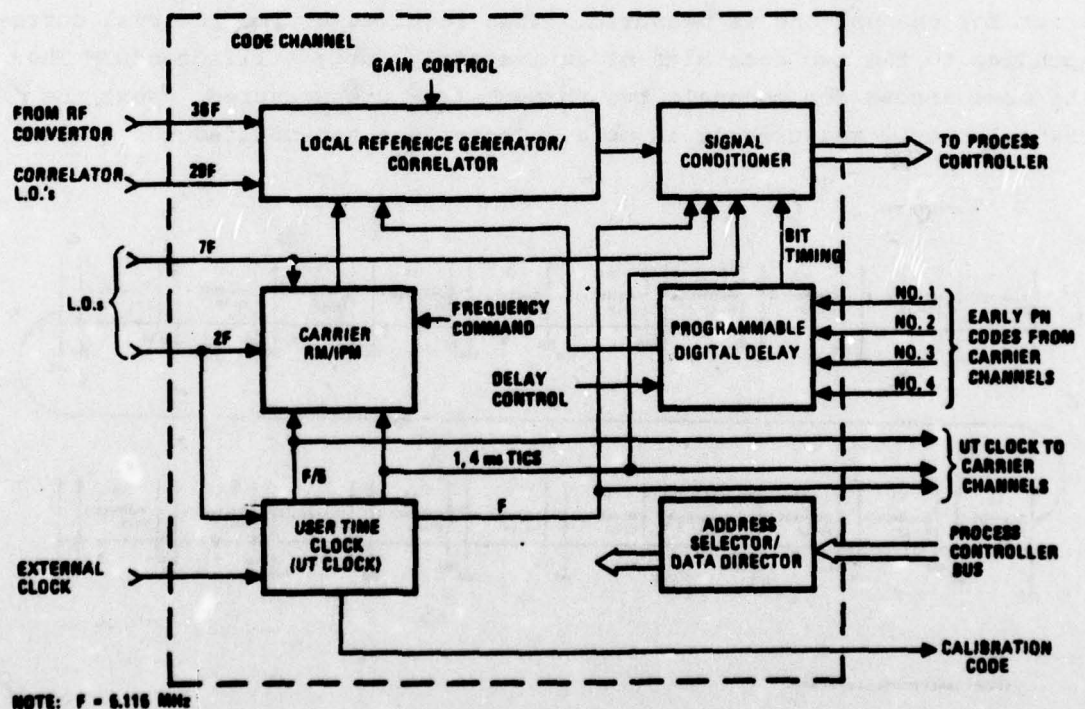


Figure 4-17. Code channel.



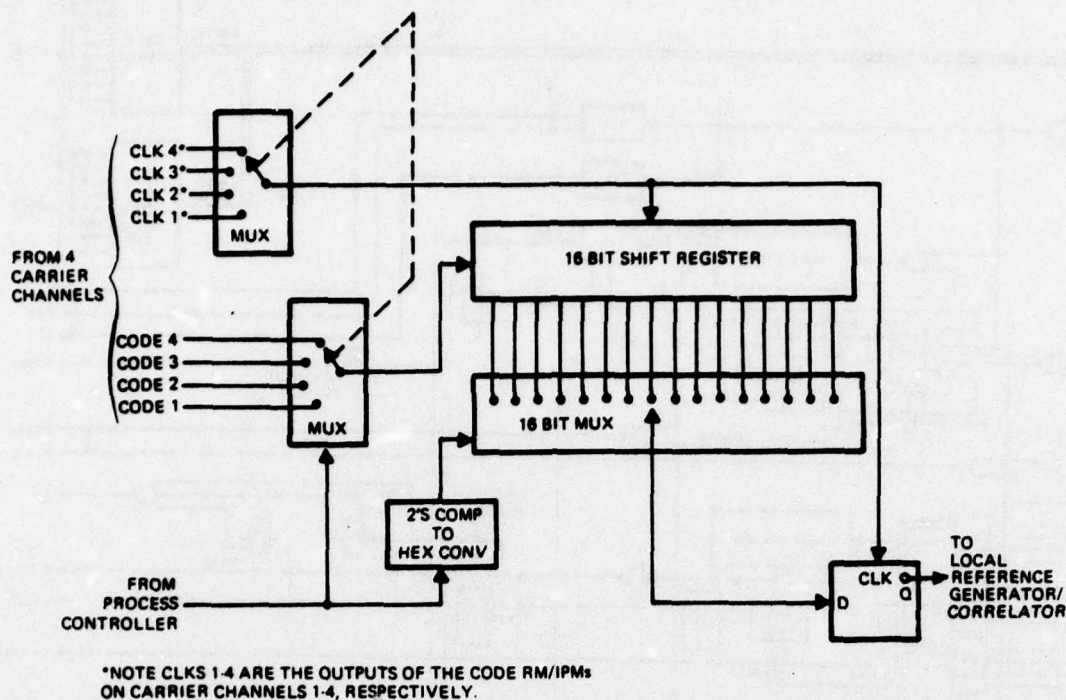


Figure 4-18. Code channel programmable digital delay.

with the incoming signal.)\* The functional operations of the user-time clock and the programmable digital delay are described in the following paragraphs.

The user-time clock provides accurate four-millisecond tics for the code generators, programmable digital delay, and signal conditioners and timing signals of frequency  $0.2F = 1023$  Hz for the carrier and code RM/IPMs.

The programmable digital delay selects the proper code and code clock (the output of the proper code RM/IPM). The code is then fed into a sixteen-bit shift register in  $1/2$  chip increments. The process controller determines the bit in the shift register, and therefore the code delay from  $-4$  to  $+4$  chips in  $1/2$  chip increments relative to the on-time estimate, that is fed to the signal conditioner. A functional block diagram of the programmable digital delay is shown in Figure 4-18.

\* The number of leads running between modules may also be a consideration.





#### 4.1.2.5 Power Supply

The power supply converts its input power taken from the aircraft power source and outputs the required voltages for the preamplifiers, receiver, signal processor and data processor. The output of the power supply is 750 watts.

#### 4.1.3 Signal Processing (Process Controller) (4-4,5,6)

The GPS receiver can operate in any one of four different modes: initialization, calibration, coordinated search, and channel independence. Figure 4-20 is a flow chart describing mode transitions. Mode selections are made by the signal processor under the supervision of the data processor.

The basic functions performed by the signal processor are receiver control, communication with the data processor, and processing of receiver signals. These three tasks are performed in all four of the operating modes, and are accomplished by means of the sixteen software components listed in Table 4-5. Some of these components operate in foreground, some in background, and some in both. All foreground processing, except during the power-up phase of initialization, is controlled by either a 1- or a 4-millisecond interrupt generated by the user time (UT) clock, depending on the mode each channel is in.

The following paragraphs describe each of the software components.

##### 4.1.3.1 Initialization

There are two initialization functions. The first occurs as a power-up sequence. It initializes the appropriate variables and constants and then enables clock and I/O interrupts.

The second function is a receiver hardware initialization sequence. This includes initializing user time from the hardware UT clock, sending initial P-code and AGC settings to the channels, setting to zero all RM-IPM's, enabling channel interrupts, and verifying thermal stability of the oscillator.

##### 4.1.3.2 UT Interrupt Processing

This function performs both user-time management and UT interrupt handling. The first of these includes setting the UT clock, either from the Handover Word (HOW) or an external reference, resetting the clock at the end of the week, and maintenance of user time in memory.

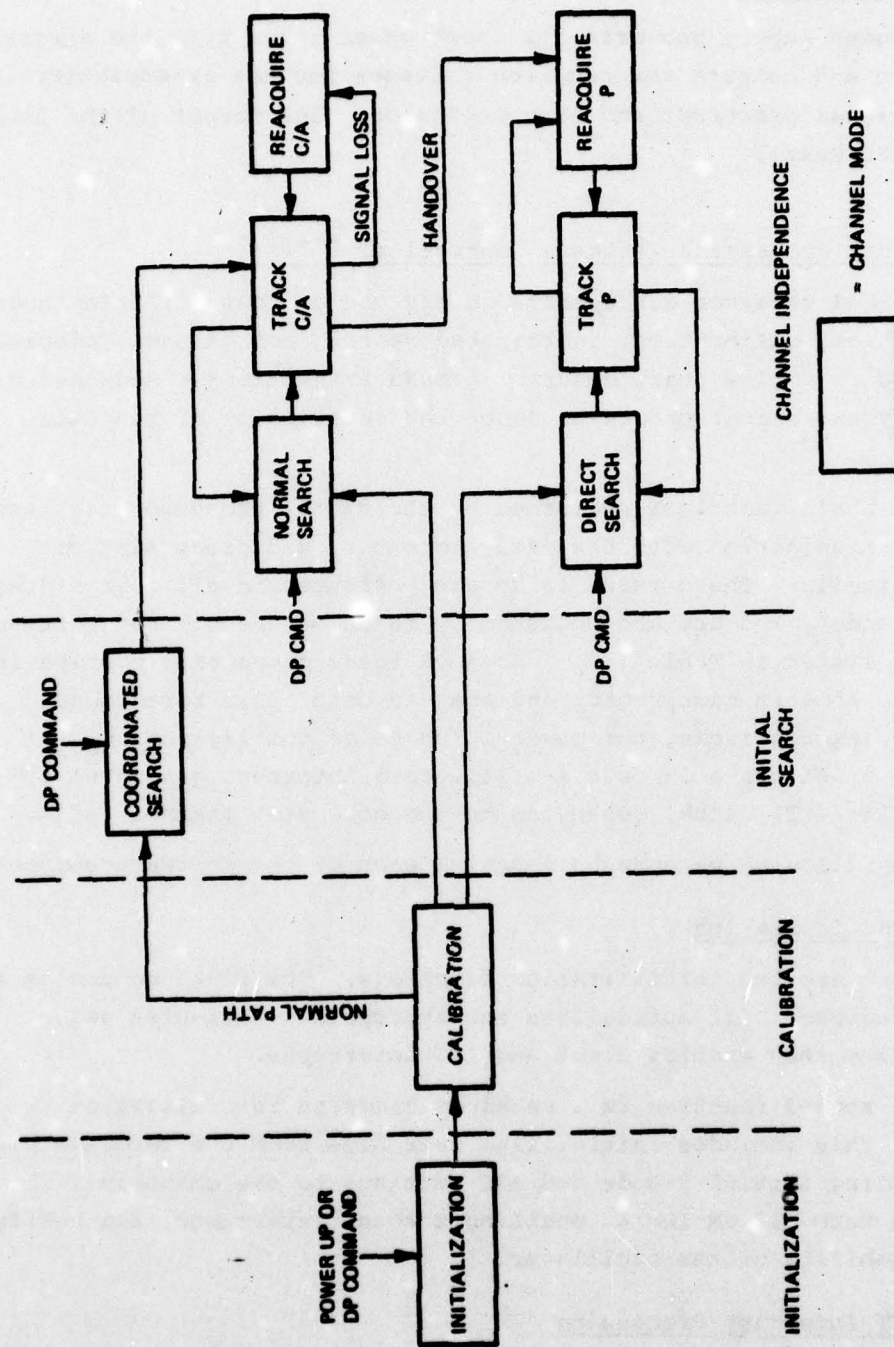


Figure 4-20. Signal processor mode transitions.



Table 4-5. GPS signal processor baseline functions.

Function	Foreground Procedure	Background Procedure
1. Initialization	x	
2. UT Interrupt Processing	x	
3. Background Loop		x
4. Fault Detection	x	
5. Calibration	x	
6. Coordinated Search	x	
7. Channel Independence	x	x
8. Data-Processor Communication	x	x
9. Channel Interrupt Processor	x	
10. Sequential Detection	x	
11. Carrier-Tracking Loop	x	x
12. Code-Tracking Loop		x
13. AGC		x
14. Satellite-Data Management		x
15. Coder Management		x
16. Measurements	x	x

NOTE: All functions except the Background Log are interrupt driven.

The interrupt handler invokes, at each UT interrupt, the sequencing and monitor function appropriate for the current receiver mode.

#### 4.1.3.3 Background Loop

This function is the dispatcher for the various background tasks. It is active whenever the processor is not engaged in handling an interrupt. Its function is to invoke various receiver and channel tasks, according to their priority.

#### 4.1.3.4 Fault Detection

This detects and reports to the data processor several types of hardware and software faults, including: hardware automatic fault indication (AFI) bits, UT timing faults, channel timing faults, diagnostics for the process controller, and memory diagnostics.

#### 4.1.3.5 Calibration

This procedure controls a sequence of functions required to align the coders and measure correction terms. The procedure also indicates faults in the receiver hardware.

Calibration is achieved by applying an L1 test signal to antenna A1 and after lock is achieved on all channels and the code loops are stable, the pseudoranges are zeroed. Then the other three combinations of frequency and antenna are used, and the pseudoranges are measured as correction constants for those configurations.

#### 4.1.3.6 Coordinated Search

This function provides an acquisition sequence which attempts to minimize the time required to achieve the first fix.

The basic strategy is to assign all five correlators (four carrier and one code) to search for the signal assigned to channel 1. The region of uncertainty is divided among the available correlators and when the signal is found, it is passed to channel 1 for C/A code tracking, while the remaining correlators search for the signal assigned to channel 2. This proceeds until all four signals are acquired.

#### 4.1.3.7 Channel Independence

Upon completion of the coordinated search, each channel is treated independently of the others, and is sequenced through signal pull-in, P-code handover, and track. If a channel loses its signal, it must reacquire it without the aid of the other correlators, using an independent (noncoordinated), P or C/A code search and reacquisition procedure.



#### 4.1.3.8 Data-Processor Communication

This function provides for processing of data transferred to and from the data processor via its DMA interface. The transfers themselves are all initiated by the data processor.

An overview of the various types of data transferred across both the receiver and the data-processor interfaces is shown in Figure 4-21. Information transferred between the data processor and process controller is divided into five groups. The direction of data transfer and a description of each group is provided in Table 4-6. The communication protocol for these transfers is shown in Figure 4-22.

#### 4.1.3.9 Channel Interrupt Processing

For each channel, this function inputs the I and Q words and performs the preprocessing appropriate to the procedure in progress for that channel, such as search, pull-in, bit synchronization, track and code-channel processing.

#### 4.1.3.10 Sequential Detection

The C/A code search in the X-set is performed by the sequential detector. The sequential detector searches in both time and frequency. A uniform-time distribution and a Gaussian-frequency distribution are assumed. The frequency search is centered about the pseudo-range rate estimate provided by the data processor. The magnitudes of the time and frequency uncertainties are also provided by the data processor. For initial C/A acquisition, these values are 1 ms and 800 Hz (1- $\sigma$ ), respectively.

The sequential detector performs a maximum-likelihood-ratio test on an approximation of the envelope of the received signal correlated with the current code and frequency settings. There are three regions in the ratio test; rejection, acceptance, and continue regions. If after a fixed number of samples, 128, the current code and frequency settings have not been rejected, the sequential detector assumes that the signal-envelope estimate is within the acceptance region. A functional block diagram of the sequential detector is shown in Figure 4-23. The inputs to the sequential detector are the one-millisecond inphase and quadrature samples,  $I_k$  and  $Q_k$ , respectively. The envelope of the correlated signal is approximated by

$$\text{env} = |I_k| + |Q_k|$$

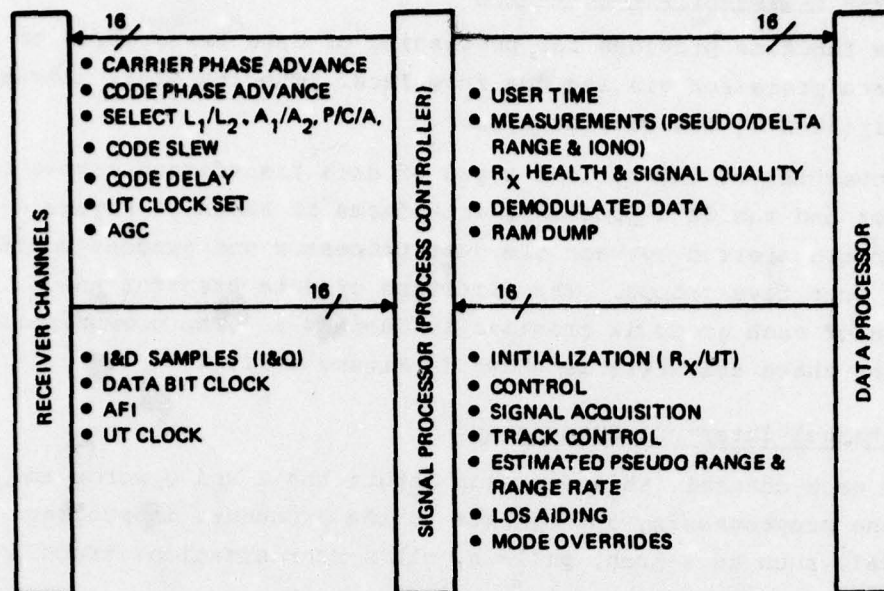


Figure 4-21. Signal processor I/O information flow.

Table 4-6. Receiver I/O data groups.

Group	Direction	Description
I	PC*to DP**	Receiver measurements and status
II	PC to DP	Satellite data, $L_1/L_2$ phase measurement
III	PC to DP	Receiver RAM dump for diagnostic
IV	DP to PC	Receiver control and channel assignments
V	DP to PC	Receiver track aiding

\*PC - Process Controller

\*\*DP - Data Processor



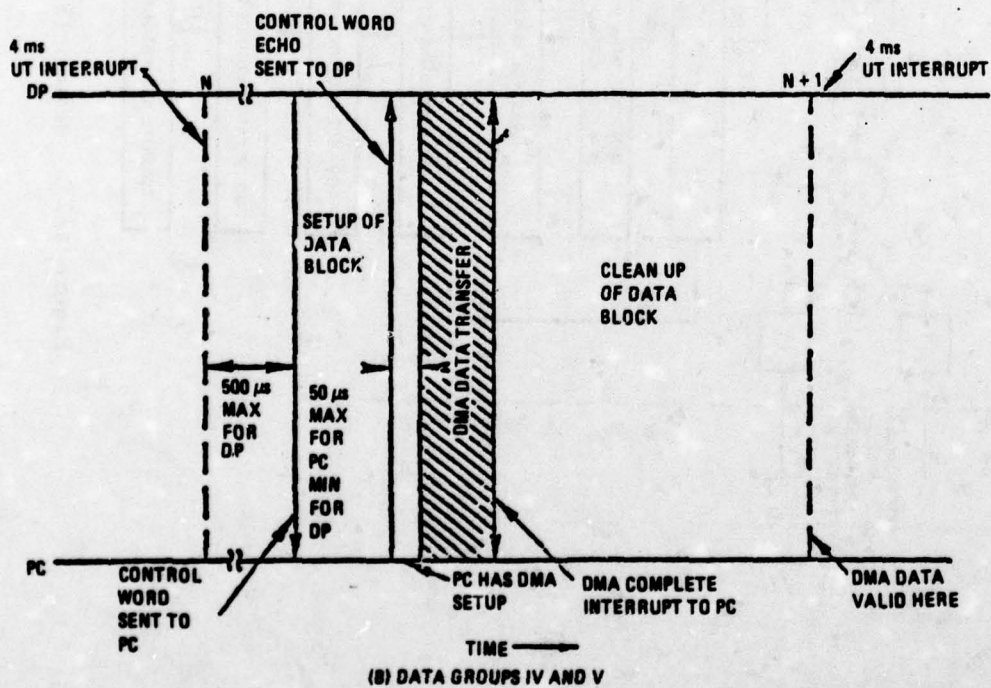
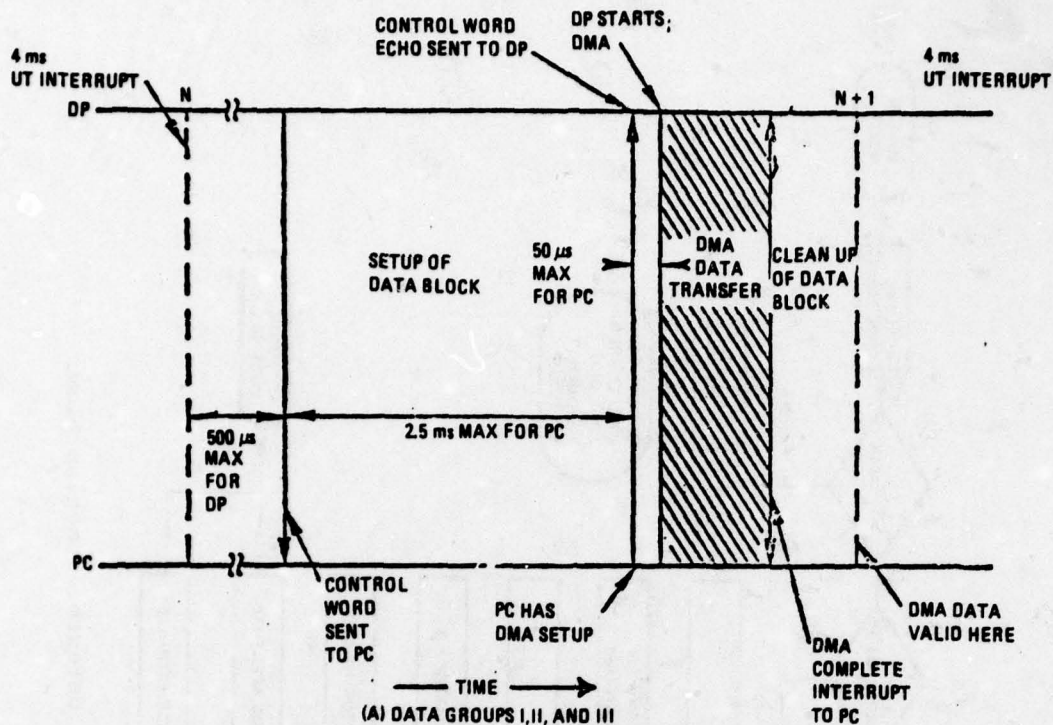


Figure 4-22. Typical timing and sequencing of DP/PC data transfer for data groups I through V.

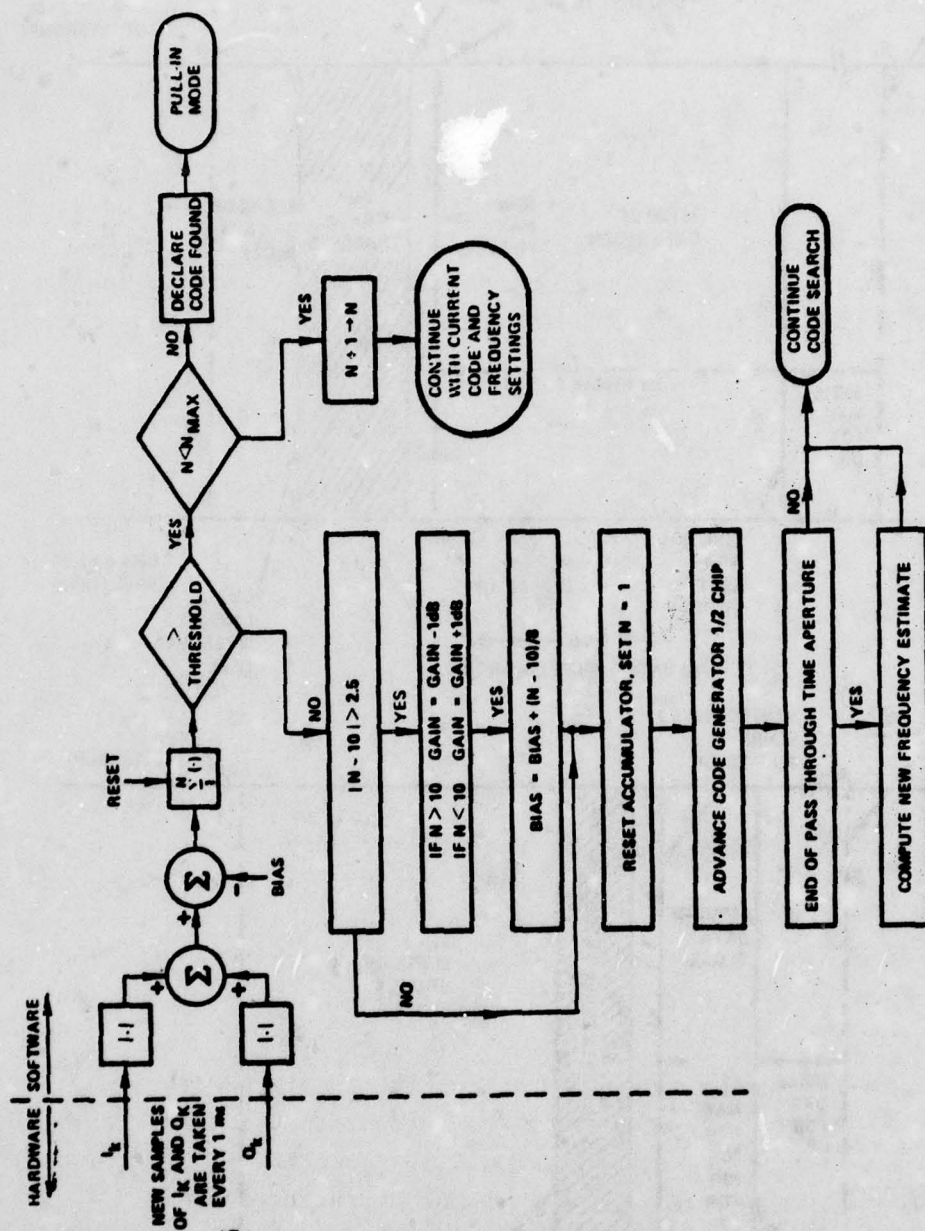


Figure 4-23. Sequential detector function flow.



A bias is subtracted from the envelope approximation. The difference is accumulated, and after each sample the accumulation is compared with the rejection threshold. (As of this writing, the rejection threshold has not been determined.) If the accumulation is below the rejection threshold the current code setting is rejected. The desired average number of samples  $\bar{N}$  accumulated prior to rejecting a code setting is ten. If the current number of samples differs from this desired average by more than 25% the gain of the automatic gain control circuitry is approximately increased or decreased by 1 dB. If this is not the case, a new bias value is determined. The new bias value is determined in the following manner

$$\text{bias} = \text{bias} + (N-10)/8$$

where N is the current number of samples taken prior to dismissal. N is then set to one and the contents of the accumulator zeroed. The code generator is then advanced by  $1/2$  a chip. After the sequential detector advances the code generator, it determines whether it has completed a pass through the time aperture. If it has, a new frequency estimate is computed and the process is repeated with the new frequency and code estimates. If not, the process is repeated with the current frequency estimate and the new code estimate.

However if the accumulation of the envelope estimates minus bias terms is greater than the rejection threshold, the number of samples in the accumulation is tested to see if it is less than the maximum number of samples,  $N_{\text{max}} = 128$ , that must be taken before code acquisition is tentatively declared. If N is less than 128, N is incremented and the process continues. If not, tentative code acquisition is declared and the channel enters the Pull-In mode.

In the Pull-In mode the receiver attempts to pull the frequency within the range of the AFC/Costas loop and to pull-in the code. The Noncoherent-Delay-Locked-Loop (NDLL) is used for code pull-in. A first-order Noncoherent Frequency-Locked Loop (NFL) operating for a predetermined time interval, 1 second, is employed to generate an estimate of the carrier frequency. At the end of the time interval the receiver switches to the AFC/Costas loop and the AFC lock indicator is monitored. If after 1 second the indicator fails to indicate lock, false code lock is declared and the code search mode is reentered. However if AFC lock is obtained, the AFC/Costas loop and the NDLL are used to track the frequency and code, respectively.

A block diagram depicting pull-in implementation is shown in Figure 4-24.

#### 4.1.3.11 Carrier-Tracking Loops

The carrier tracking loop filter implementation depends on the signal-to-noise ratio. Under normal tracking conditions, i.e., good signal-to-noise ratios, the receiver employs a Costas loop for carrier tracking. A block diagram depicting the implementation of the Costas loop in the X-set is shown in Figure 4-25.

In situations where Costas lock is not possible (e.g., under high jammer-to-signal conditions or during initial acquisition), the X-set receiver must estimate the carrier frequency as well as possible so that the carrier loop can provide accurate velocity aiding information to the code loop. For these purposes Automatic Frequency Control (AFC)/Costas loops are used. For acquisition, a first-order AFC and a second-order Costas loop are used. A block diagram of this implementation in the X-set is shown in Figure 4-26. Except for acquisition, a second-order AFC and a third-order Costas loop are used. A diagram depicting this implementation is shown in Figure 4-27.

In the preferred implementation of this receiver, the carrier-tracking loops are implemented in hardware, substantially reducing the software load.

Figures 4-24, 4-25, and 4-26 show the loop implementations for Costas, AFC/Costas, and acquisition modes, respectively.

#### 4.1.3.12 Code-Tracking Loops

The signal processor controls the time-sharing of the code channel. This channel is cycled between each of the four carrier channels, making estimates of code errors. This error estimate is then used to generate a phase correction which is applied to the carrier channels code rate multiplier.

The implementation of the code-tracking loop is shown in Figure 4-28 and the sequencing schedule for the code channel is shown in Figure 4-29.

#### 4.1.3.13 AGC

Due to variations in channel correlator gains, signal levels, and non-linearity of the software selectable attenuator, each channel



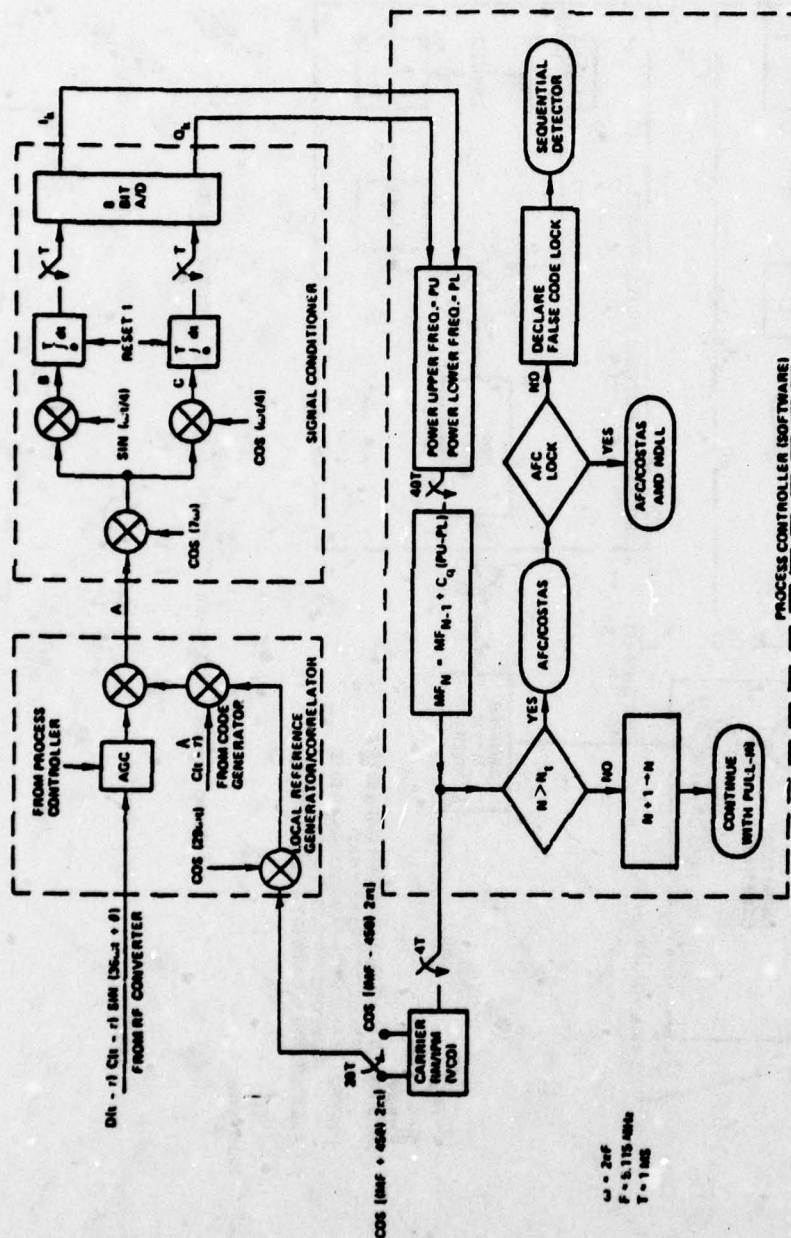


Figure 4-24. Noncoherent frequency-locked loop used in X-set receiver for frequency pull-in.

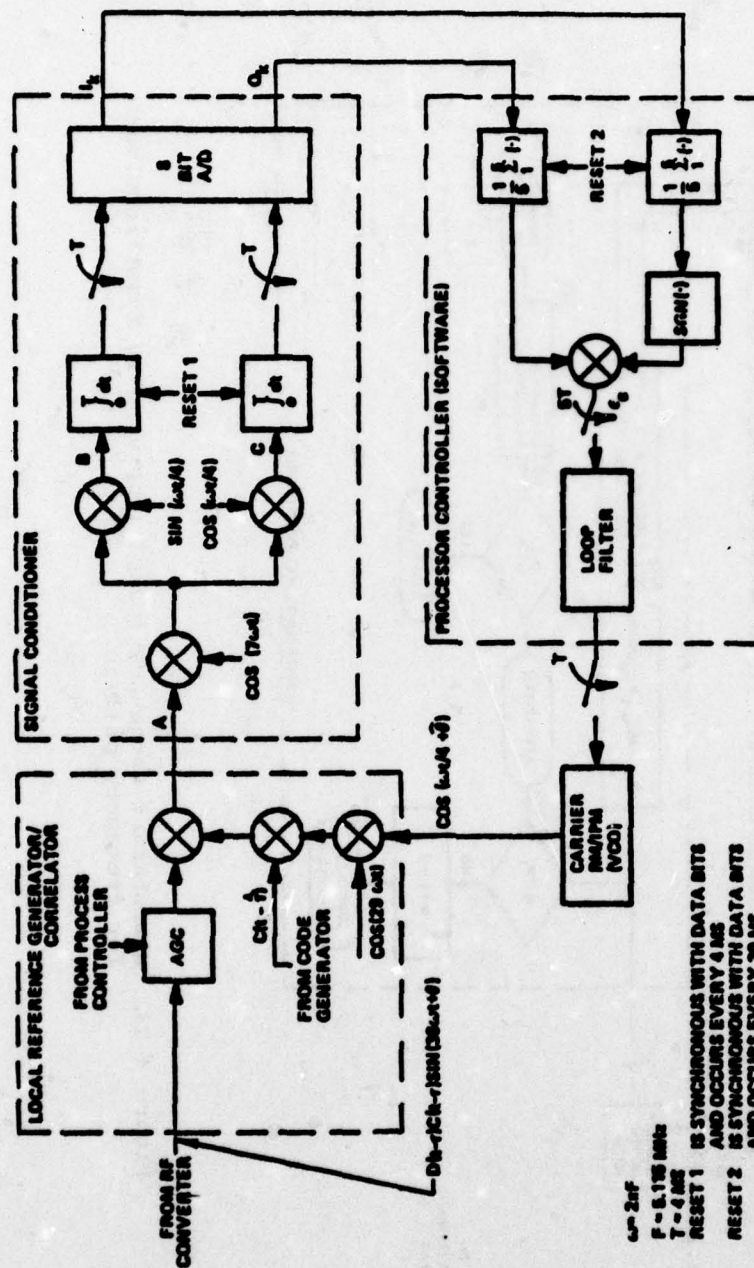


Figure 4-25. X-set Costas loop implementation.

SB 270 002A



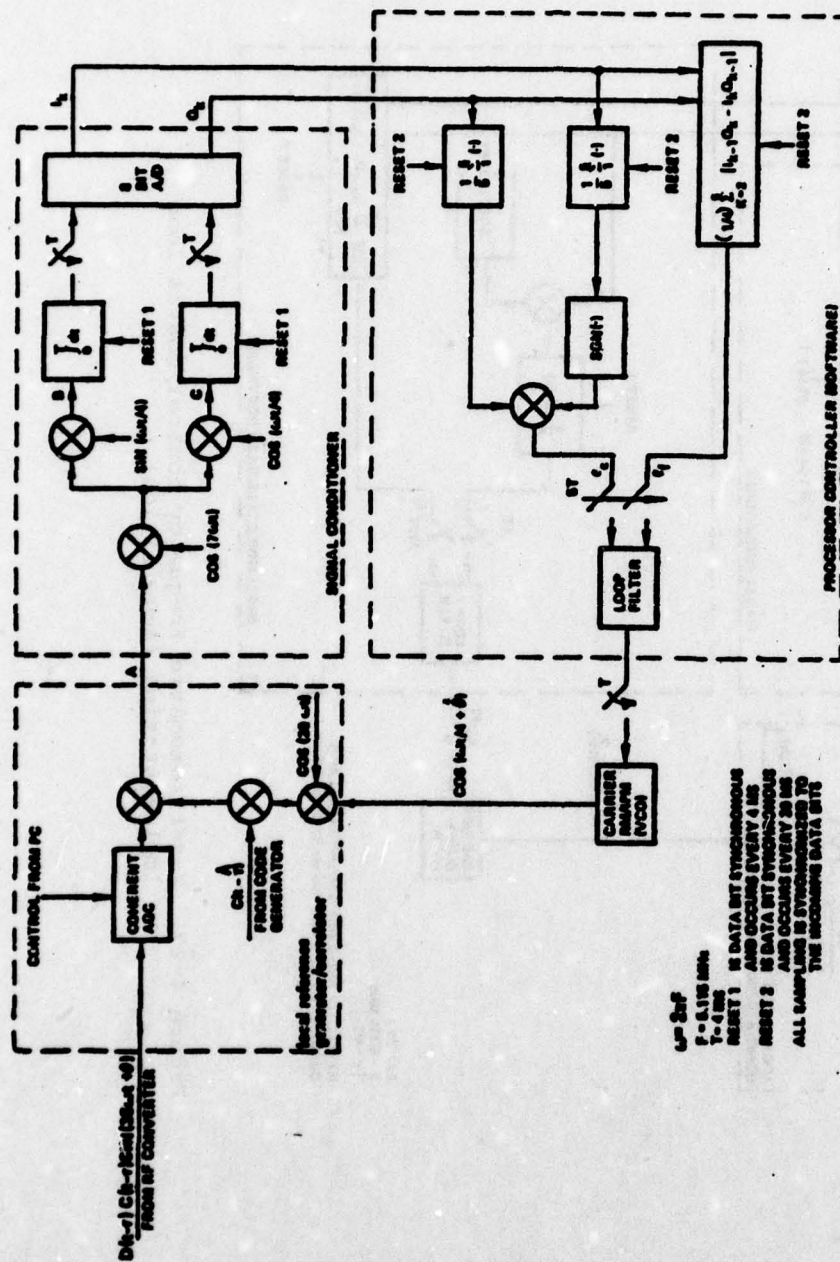


Figure 4-26. X-set automatic frequency control/Costas loop implementation (HOBYT).





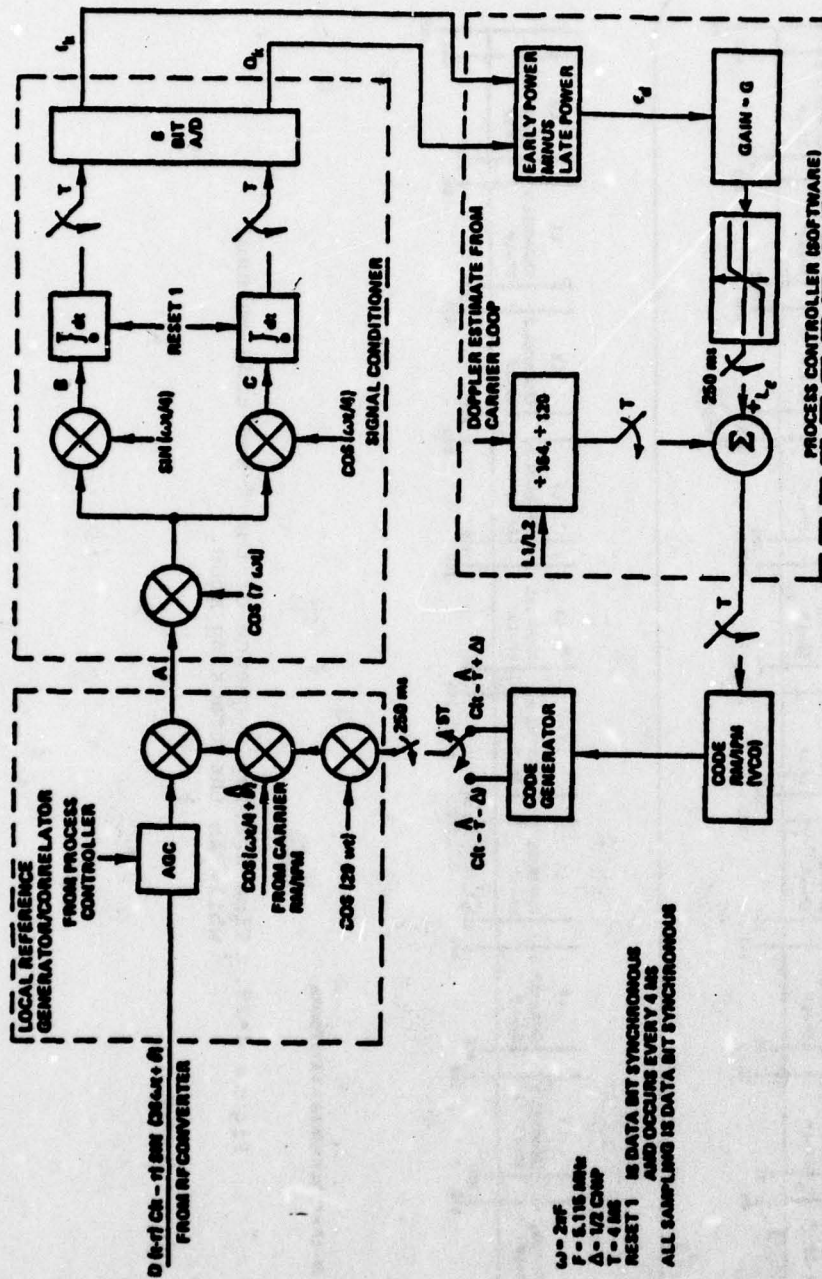


Figure 4-28. X-set code tracking loop.

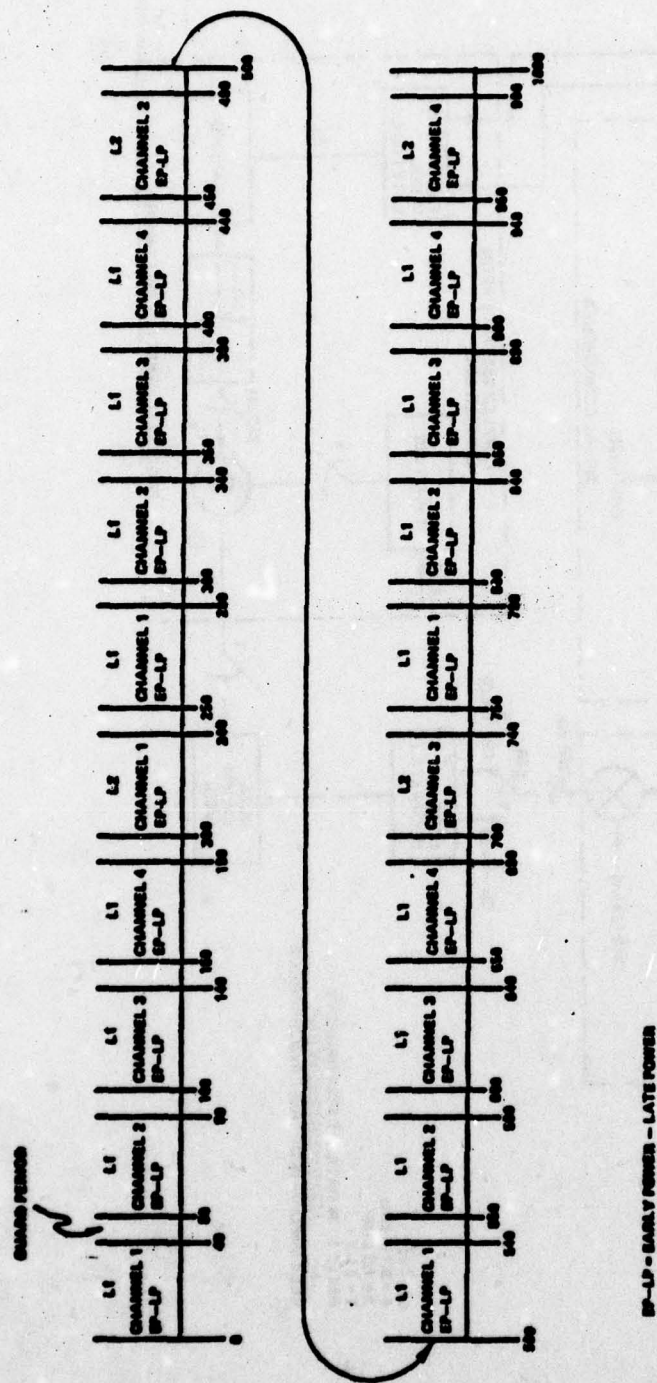


Figure 4-29. Time sharing sequence of the X-set code channel while in the tracking mode.



requires an independent AGC loop. It is the function of the signal processor to control this gain.

The particular gain setting for any channel is mode-dependent. For instance, during initialization, the setting is used for noise-floor measurement, and during track it is used to ensure Costas loop operability. Table 4-7 shows how AGC is controlled during various signal processor operations.

Table 4-7. AGC requirements.

AGC Mode	Requirements	Measurement Parameter	Adjustment Interval	Time Constant (s)
Initialize	Set AGC within 1 dB to assure proper operation of sequential detect	$1 + Q$	20 ms	0.2
Search C/A ACQ Direct P-ACQ C/P Handover	Adjust AGC to maintain constant rate search	Dismissal count	1 s	N/A
Pull-in	Gross adjustment of signal plus noise to pre-scale for costas	Low frequency power plus high frequency power	40 ms	0.1
Early Late	Fine adjustment of signal plus noise as the code centers	Early power plus late power	90 ms	0.1
AFC	Adjust signal plus noise	$1 + Q$	20 ms	1.0
Costas Phase Lock	Adjust gain to correlator for nominal coherent signal	1	20 ms	1.0
Aiding	Adjust signal plus reduced noise to costa operation level	NBP-WBF	40 ms	10.0

#### 4.1.3.14 Satellite-Data Management

This function synchronizes each carrier channel to the 20-millisecond data stream, demodulates the data bits, combines them into 30-bit data words, checks parity, sets the hardware clock from the HOW, and synchronizes to the preamble of the telemetry word of a sub-frame. This data is transmitted to the data processor via the Data-Group II.

#### 4.1.3.15 Coder Management

This function controls the code generators in each of the four carrier channels. The codes appropriate to each satellite are sent to the channel and the codes are slewed an amount derived from estimated range, user time, and HOW data, if acquiring P-code, or they are slewed according to a predetermined search algorithm if acquiring C/A code.

#### 4.1.3.16 Measurement Processing

The signal processor provides for measurement of pseudorange, delta range, and ionospheric effects, and for estimation of the signal-to-noise ratio for each channel.

Pseudorange is simply the accumulated phase difference between the code and user time since calibration, while delta range is the sum of velocity and phase corrections to the channel since the last measurement. Ionospheric effects are estimated by an  $L_1 - L_2$  offset measurement.

#### 4.1.4 Data Processing

Figure 4-30 is a functional block diagram of the GPS data processor showing data flow and functional relationships between the major functions. The bombing function which is performed by the current X-Set implementation has been deleted in order to reflect operational requirements instead of the current development and evaluation requirements.

The following paragraphs describe the processing performed by each function in greater detail. Complete descriptions of all processing are given in References 4-7 through 4-11.

##### 4.1.4.1 Executive And Service

This area is comprised of the subfunctions listed below:

- 1 - Execution control
- 2 - Interface maintenance



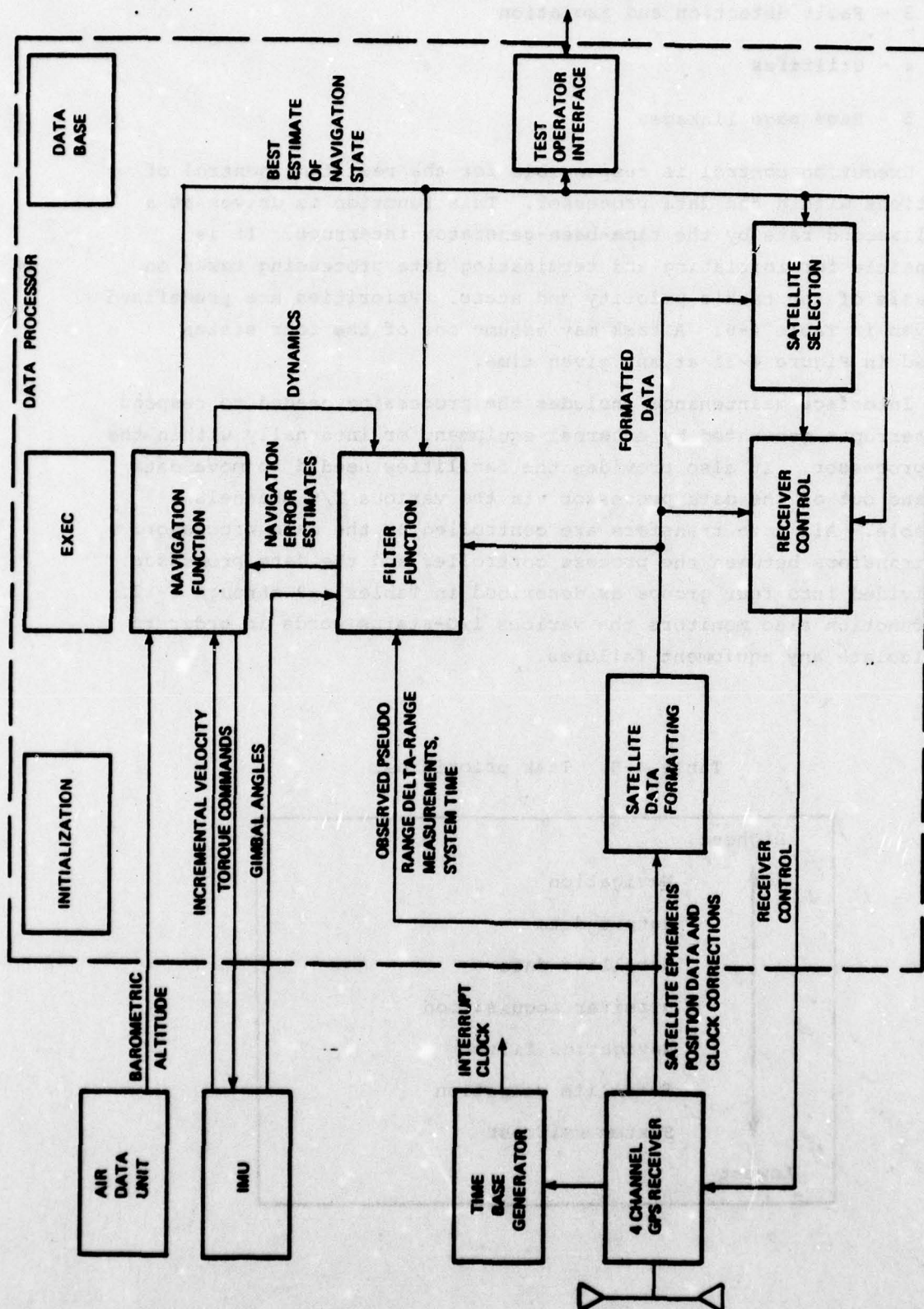


Figure 4-30. GPS data processing block diagram.

3 - Fault detection and isolation

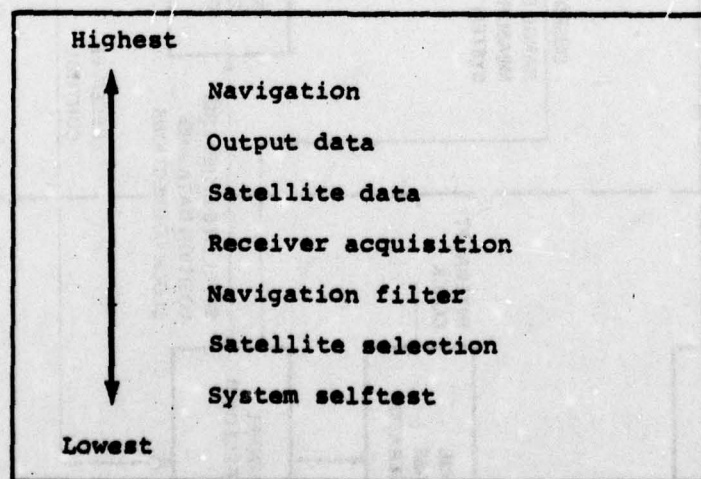
4 - Utilities

5 - Base page linkages

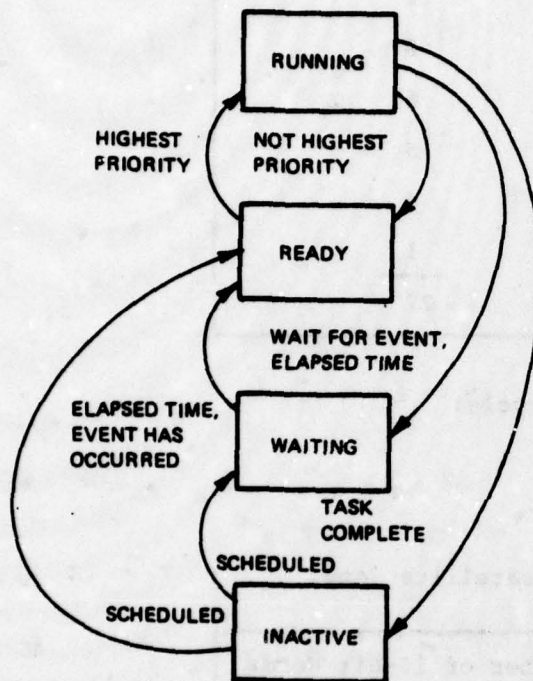
Execution control is responsible for the real-time control of operations within the data processor. This function is driven at a 4-millisecond rate by the time-base-generator interrupt. It is responsible for initiating and terminating data processing tasks on the basis of the task's priority and state. Priorities are predefined as given in Table 4-8. A task may assume one of the four states defined in Figure 4-31 at any given time.

Interface maintenance includes the processing needed to respond to interrupts generated by external equipment or internally within the data processor. It also provides the facilities needed to move data into and out of the data processor via the various I/O channels available. All data transfers are controlled by the data processor. Data transfers between the process controller and the data processor are divided into four groups as described in Tables 4-9 through 4-12. This function also monitors the various I/O-status words in order to help isolate any equipment failures.

Table 4-8. Task priorities.







**NOTE:**

ALL TASKS ARE IN ONE OF FOUR STATES:

STATE	DEFINITION
EXECUTING	THE TASK BEING PROCESSED BY THE NDP
READY	A PRIORITY ARRANGED QUEUE OF TASKS THAT ARE READY TO EXECUTE
WAITING	A QUEUE OF TASKS WAITING FOR A TIME INCREMENT OR OTHER EVENT TO BE MOVED TO THE READY QUEUE
INACTIVE	ALL OTHER TASKS

Figure 4-31. Task states.

Table 4-9. Data group I, range data.

Description	Number of 16 Bit Words
User time of day	2
Channel 1 range	2
Channel 1 delta range	2
Channel 1 health and quality	2
Channel 2 - same as Channel 1	6
Channel 3 - same as Channel 1	6
Channel 4 - same as Channel 1	6
Receiver quality	<u>1</u>
Total	27

Frequency - 1/5 hertz (every Kalman cycle)

Process Controller to Data Processor

Table 4-10. Data Group II, satellite data.

Description	Number of 16-Bit Words
Channel 1 satellite data	20
Channel 2 satellite data	20
Channel 3 satellite data	20
Channel 4 satellite data	<u>20</u>
Total	80

Frequency = 1/5.4 hertz

Process Controller to Data Processor



Table 4-11. Data Group IV, receiver assignment and control.

Description	Number of 16-Bit Words
General control and receiver time initialization	3
Acquisition control (1 word per channel)	4
Track control (1 word per channel)	4
Estimated range (2 words per channel)	8
Estimated delta range (1 word per channel)	4
Total	<hr/> 23

Frequency - as required

Data processor to Process Controller

Table 4-12. Data Group V, code loop aiding.

Description	Number of 16-Bit Words
Time and Status	1
Delta range for Channel 1	1
Delta range for Channel 2	1
Delta range for Channel 3	1
Delta range for Channel 4	1
Total	<hr/> 5

Frequency = 10 hertz

Data Processor to Process Controller

In addition to the fault isolation within the various interfaces, there is also a test for the entire data processor memory. Each memory location is first saved and then three test patterns are written to and read from the location. The original data is then restored. The memory test is performed in a manner which prevents other tasks from accessing the memory location being tested. Status flags from other functions are also monitored for error conditions.

The utilities include those facilities such as the FORTRAN library which may be used by many of the tasks within the data processor. Because of the real-time nature of GPS data processing, it is quite possible that these facilities must be provided in a reentrant form.

The last subfunction to be included here is referred to as base page linkage. Because the computer selected for the X-set data processor can directly address only 2,048 words of memory, 1024 words are reserved to hold addresses. A task must reference this base page in order to access an address which it was unable to reference directly.

#### 4.1.4.2 Navigation

This function is responsible for propagating the user navigation state in IMU-aided and unaided (no IMU) configurations. It also controls the scheduling of the filter, receiver control, and satellite selection tasks. Incremental velocity is provided at a ten hertz rate when the IMU is available and the navigation routines are also initiated at this rate. The navigation software provides for IMU alignment as well as navigation in a local-level wander-azimuth coordinate frame. The local-level wander azimuth direction-cosine matrix utilized in this mechanization is reorthogonalized once every second. The gyro-torque commands required to maintain the IMU in a local-level orientation are computed and issued from this task. When the IMU is available, receiver-aiding information is provided to the X-set receiver channels. Barometric altimeter data is used for vertical-channel damping when available. The navigation function is also responsible for providing waypoint steering when that mode is requested by the operator.

#### 4.1.4.3 Navigation Filter

The filter task is not executed at a fixed rate. Depending on the total processing load, its execution period can vary from approximately three seconds to twelve seconds. Twelve state variables are utilized as shown in Table 4-13. The filter states vary depending on which IMU mode, aided or unaided, has been selected. The filter



Table 4-13. Filter state variables.

Filter State Variables	Aided (with IMU)	Unaided (without IMU)
$x_1$	Estimated angle misalignments, about X and Y axes, between computed and true tangent plane axes (can be thought of as horizontal position divided by radius)	Identical to aided
$x_2$		
$x_3$		
$x_4$	Estimate of error in open-loop altitude state	
$x_5$		
$x_6$		
$x_7$	Estimated of clock phase error	
$x_8$	Estimate of clock frequency error	
$x_9$	Estimate of error in external altitude reference	
$x_{10}$	Estimate of computer-to-platform misalignment angles	
$x_{11}$		
$x_{12}$		
		Estimates of components of kinematic acceleration error

function is responsible for the following processing:

- (1) Sets up the transition and process-noise matrices.
- (2) Propagates filter statistics over the measurement period.
- (3) Computes predicted observables.
- (4) Computes Kalman gains.
- (5) Updates error-state estimates.
- (6) Maintains filter-measurement statistics.

The navigation-filter design is based on a fast triangular formulation of a square-root filter. This implementation reduces core-storage requirements, provides a fast execution cycle, and reduces numerical range problems associated with the computation of the filter-covariance matrix. Figure 4-32 shows the relationship between the navigation and filter functions when IMU data is provided.

#### 4.1.4.4 Satellite Selection

This function operates at a relatively low priority and is responsible for determining which four satellites or ground transmitters are to be tracked by the receiver. Utilizing almanac data, vectors to all potentially visible satellites are computed. Four satellites are selected dependent upon atmospheric effects, mission requirements, satellite health and expected duration-of-satellite visibility. Following the initial selection, the performance index of the constellation currently in use is constantly compared to the performance indices of candidate-replacement constellations. New satellites are selected as a result of necessity or the potential for improved performance. Operator specified transmitters are always selected for use even though "better" constellations may be available.

#### 4.1.4.5 Receiver Control

This function is executed whenever the receiver is trying to acquire the signal from a new satellite or reacquire a lost signal from an old satellite. This function is responsible for selecting the appropriate carrier frequencies, acquisition codes and carrier bandwidths and passing these commands to the receiver. It also reconfigures the



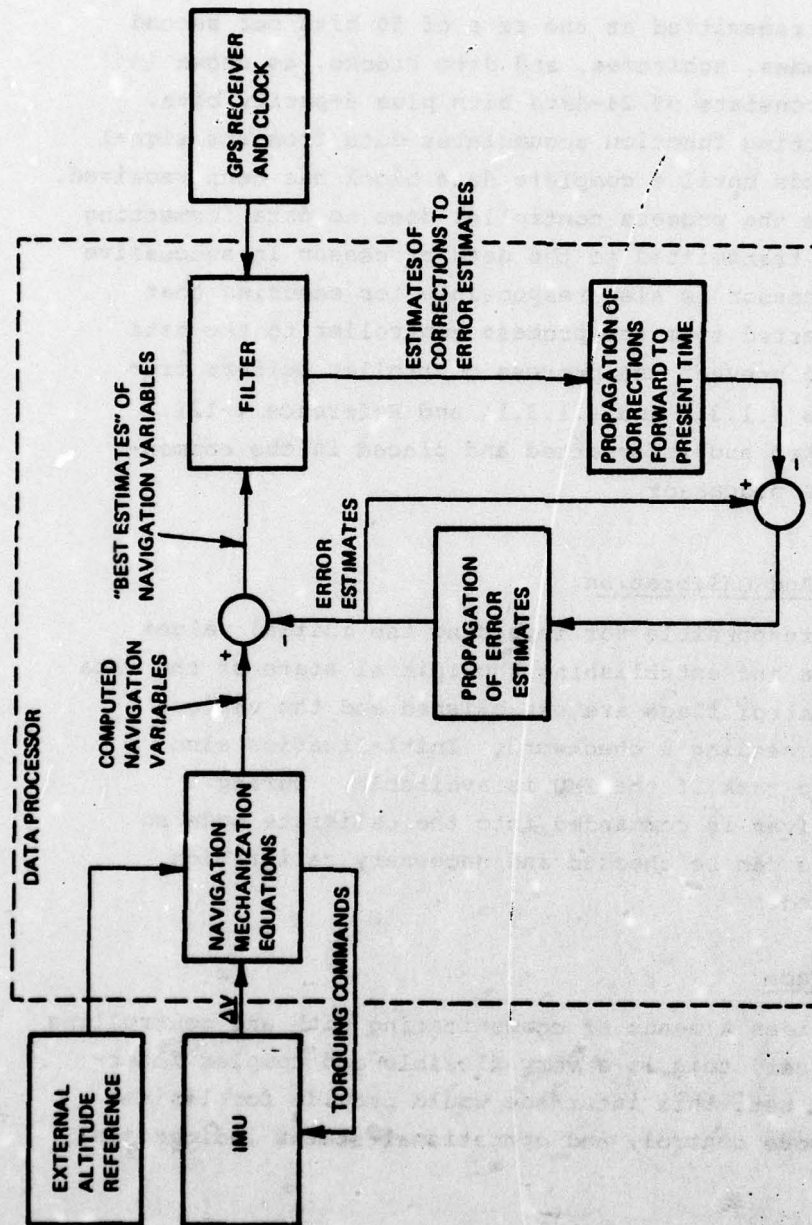


Figure 4-32. Aided navigation.

receiver should problems occur during acquisition. This function passes pseudo-range and delta-range rate uncertainties to the receiver to aid in signal acquisition. This function also controls the antenna selection.

#### 4.1.4.6 Satellite-Data Formatting

Satellite data is transmitted at the rate of 50 bits per second and is organized into frames, subframes, and data blocks, as shown in Table 4-14. Each word consists of 24-data bits plus 6-parity bits. The satellite-data formatting function accumulates data from the signal processor every 5.4 seconds until a complete data block has been received. This is necessary because the process controller does no data formatting so duplicate data may be transmitted to the data processor in successive transfers. The data processor is also responsible for ensuring that satellite data is transferred from the process controller to the data processor often enough to prevent the process controller buffers from overflowing (see Sections 4.1.3.8 and 4.1.3.14 and Reference 4-12). Error-free data is unpacked and reformatted and placed in the common-data base within the data processor.

#### 4.1.4.7 Initialization And Calibration

Initialization is responsible for inputting the initial values into the common-data base and establishing the initial state of the data processor. Necessary control flags are established and the various interfaces are tested by sending a checkword. Initialization also schedules the IMU starcup task if the IMU is available. During initialization, the receiver is commanded into the calibrate mode so that receiver performance can be checked and necessary calibration parameters can be computed.

#### 4.1.4.8 Operator Interface

This function provides a means of communicating with and controlling the receiver. In the X-set, this is a very flexible and complex interface. In an operational set, this interface would provide for limited system initialization, mode control, and operational-status indications.



Table 4-14. Data frame.

Word No.	Data Block I	Data Block II		Special Message	Data Block III				
	Subframe No. 1	Subframe No. 2	Subframe No. 3	Subframe No. 4	Subframe No. 5				
1	TLM	TLM	TLM	TLM	TLM				
2	HOW	HOW	HOW	HOW	HOW				
3	Spares	Ephemeris Data	Ephemeris Data	TBD	Satellite Almanac				
4									
5	Ionospheric Data								
6									
7	Clock Corrections								
8									
9									
10									

Subframe 1    Subframe 2   Subframe 3   Subframe 4    Subframe 5

#### 4.1.4.9 Data Base

The data base contains the information which is common to more than one program module in the data processor.

#### 4.1.5 Processing Requirements

The following paragraphs describe the memory requirements and processing power needed to provide the GPS signal and data processing capabilities as described in the preceding paragraphs.

##### 4.1.5.1 Signal Processing (Process Controller)

The process-controller computer was developed by Magnavox and is described in considerable detail in Reference 4-4. It was designed to emulate a subset of the Hewlett Packard HP-21MX instruction set. Table 4-15 provides a summary of the salient characteristics of the process controller.

Table 4-15. Process controller characteristics.

Microprogram Memory	512 words, 32 bits wide
Micro-instruction cycle time	542 nanoseconds
Emulation type	HP21MX (Hewlett-Packard)
Arithmetic	Two's complement
Data word lengths	8, 16 and 32 bits
Data types	Fixed point
General Registers	A/B accumulators X/Y index
Memory size	12,288 words, 16 bits, ROM 2 048 words, 16 bits, RAM
Memory cycle time	684 nanoseconds
DMA rate	729K words
Interrupts	10 vectored
Direct addressing	2,048 words (base and current page)

#### 4.1.5.1.1 Memory Requirements

Table 4-16 gives the memory requirements by major function for the baseline process controller. Total memory required to implement these functions was 14,600 bytes (8 bit). All software was coded in assembly language.

#### 4.1.5.1.2 Processing Load

It was not possible to obtain the processing load as a percentage of machine utilization for each of the major process controller functions. However, percentage loads were obtained for various operating modes and are given in Table 4-17. The throughput of the process controller in terms of KOPS (thousands of operations per second) is calculated in Table 4-18 with the result used to calculate processing load



Table 4-16. Process controller memory requirements.

Function	Memory Requirements (8-bit bytes)
Initialization	500
Calibration	1,450
UT Interrupt Processing	500
Fault Detection	500
Coordinate Search	600
Channel Independence	600
Interface with Navigation Processor	1,300
Channel Interrupt and Sequential Detection	2,100
Carrier Tracking	500
Code Tracking	1,300
AGC	550
Satellite Data Management	950 plus *
Coder Management	1,650
Measurement Processing	*
Total	14,600

\* Included in other functions.

Table 4-17. Processing load for operational modes.

Function	Load* (%)
Search	15.5
Pull-In and Bit Sync	49.5
AFC	13.5
Track	17.0

\*Per channel

Table 4-18. Process controller KOPS calculations.

Instruction	Execution Time (μs)	Kops	Standard Aviation Mix (%)	Weighted Kops	DIS* Mix	Weighted Kops
Load/Store	3.28	304.9	45	137.2	36	109.8
Add/Subtract	3.07	325.7	9	29.3	14	45.6
Multiply/Divide	12.31	81.2	5	4.1	7	5.7
Shift	4.45	224.7	5	11.2	4	9.0
Logical	2.22	450.5	5	22.5	8	36.0
Test and Branch	4.28	233.6	30	70.1	30	70.1
I/O	2.39	418.4	1	4.2	1	4.2
Totals				273.6		280.4

\*Digital Integrating Subsystem

Average = 279.5

in Table 4-19. The worst-case processing load is 280.4 KOPS which occurs when three channels are tracking and the fourth channel performing pull-in and bit synchronization.



Table 4-19. KOPS load for operational modes.

Function	Load* (KOPS)
Search	43.3
Pull-In and Bit Sync	138.4
AFC	37.7
Track	47.5

\* Per channel

#### 4.1.5.2 Data Processing

The following paragraphs outline the processing requirements in terms of memory and throughput for the GPS Baseline Set. The Hewlett-Packard HP-21MX was the data processor computer used in the X-set. Most of the software was coded in a FORTRAN-like language which gives rise to some inefficiencies in memory usage and processing load.

##### 4.1.5.2.1 Memory Requirements

Table 4-20 gives a breakdown of memory usage by major function. The bombing function which is a part of the existing X-set has been deleted (approximately 5350 bytes) and the Operator Interface has been changed from a flexible test and evaluation function (10750 bytes) to an operational cockpit-type interface (approximately 750 bytes). The total memory need for the baseline X-set thus becomes 71,650 bytes. Because the HP-21MX is capable of addressing only 64K bytes of core, a memory mapping scheme was employed to extend the available memory from 64K bytes to 96K bytes. Core was divided into three 16K-word (32K-byte) sectors with certain functions forced to reside in particular sectors as shown in Table 4-21.

##### 4.1.5.2.2 Processing Load

Table 4-22 lists the salient features of the HP-21MX as used in the X-set. In order to provide a measure of the throughput of this machine, the KOPS (thousands of operations per second) figure is calculated in Table 4-23 for two instruction mixes. These numbers provide

Table 4-20. Baseline memory requirements.

Function	Memory Requirement (bytes)
Executive and Service	11,600
Navigation and Alignment	11,650
Navigation Filter	13,550
Satellite Selection	5,950
Receiver Control	7,450
Satellite Data Formatting	4,250
Initialization, Calibration, Miscellaneous	5,600
Operator Interface	750
Data Base	10,850
Total	71,650

Table 4-21. Memory mapping.

Item	Logical Address*	Physical Address*
All User Programs	0-16k 16-32k	0-16k 32-48k
Dual Channel Port Controllers	0-32k	0-32k
System Software	0-32k	0-32k

\* Word addresses

a measure of throughput which is somewhat independent of the computer and will be used in later paragraphs when developing the processing requirements for various integrated designs. The total processing load for the X-set is approximately 80% of the available machine time. It was not possible to obtain a measured breakdown of the total processing load by functions for the X-set. However, estimates of the breakdown have been made based on information provided by Intermetrics, Texas Instruments and Magnavox and are given in Table 4-24. Table 4-25 gives the same breakdown in terms of KOPS per function.



Table 4-22. HP21 MX characteristics.

Word size	16 bits plus parity bit
Word formats	Single/double precision, fixed and floating point
Number of instructions	128
Number of registers	2 accumulators 2 index registers
Page size	1,024 words
Memory map organization	3 16k words regions
Direct addressing	2 pages (base plus current)
Indirect addressing	32,768 words
Memory cycle time	650 nanoseconds
Number of I/O channels	9 (standard)
Number of DMA channels	2 dual channel port controllers (DCPC)
Number of memory ports	1
Registers/channel	2 (word count, address)
Max. block size	32,768 words
DMA priority	Highest - DCPC #1 Middle - DCPC #2 Lowest - CPU
Max. DMA transfer rate	616,666 words/second (both channels combined)
Interrupt structure	Multilevel vectored priority
Power fail interrupt	Highest priority interrupt
Parity fail interrupt	Second highest priority interrupt

AD-A065 136

CHARLES STARK DRAPER LAB INC CAMBRIDGE MA

F/G 1/3

AN INTEGRATED FAULT-TOLERANT AVIONICS SYSTEM CONCEPT FOR ADVANC--ETC(U)

N00019-78-C-0572

UNCLASSIFIED

R-1226

NL

4 OF 4

AD  
A03513



END  
DATE  
FILMED

4 -79

DDC



Table 4-23. Throughput for GPS data processor.

Instruction	Execution Time (μs)	KOPS	Standard Aviation Mix (%)	Weighted KOPS	DIS <sup>†</sup> Mix (%)	Weighted KOPS
Load/Store	3.44	290.7	45	130.8	36	104.7
Add/Subtract	20.72*	48.3	9	4.3	14	6.8
Multiply/Divide	32.65**	30.6	5	1.5	7	2.1
Shift	4.79	208.8	5	10.4	4	8.4
Logical	2.0	500.0	5	25.0	8	40.0
Test and Branch	3.73	268.1	30	80.4	30	80.4
I/O	3.24	308.6	1	3.1	1	3.1
<b>TOTALS</b>				<u>255.5</u>		<u>245.5</u>
				Average = 250.5		

\* 50% floating point

\*\*80% floating point

<sup>†</sup> Digital Integrating Subsystem

Table 4-24. Processing load estimates.

<u>Function</u>	<u>Processing Load (%)</u>
Executive and Service	25
Navigation	30
Navigation Filter	10
Satellite Selection	5
Receiver Control	5
Satellite Data Formatting	5
Initialization, Calibration, Miscellaneous	-
Operator Interface	-
Data Base	-
	<hr/>
TOTAL	80

Table 4-25. GPS processing load.

<u>Function</u>	<u>Processing Load (kops)</u>
Executive and Service	62.6
Navigation	75.2
Navigation Filter	25.1
Satellite Selection	12.5
Receiver Control	12.5
Satellite Data Formatting	12.5
Initialization, Calibration, Miscellaneous	Negligible
Operator Interface	Negligible
Data Base	-
	<hr/>
Total	200.4



## LIST OF REFERENCES

### SECTION 4

- (4-1) Stonestreet, William M., A Functional Description of the NAVSTAR GPS Receiver Model X, CSDL Report R-981, Volume 1, 26 April 1976, Revised February 1977.
- (4-2) NAVSTAR Global Positioning System (GPS), Concept of Operations, Approved by B. Parkinson, Col., USAF, Deputy for Space Navigation Systems.
- (4-3) Student Handbook -- GPS User Equipment Orientation Course, Magnavox APD.
- (4-4) Computer Program Development Specification for the Set X Signal Processing Software (X-SPS). CP-US-300
- (4-5) GPU User Equipment (UE) Orientation Course - Student Handbook. MX-125-C-US-7701
- (4-6) A Functional Description of the NAVSTAR GPS Receiver Model X, Final Report for SAMSO Contract F04701-75-C-0212.
- (4-7) Computer Program Development Specification for the GPS X User Set (Aided) of the NAVSTAR GPS User Equipment Segment, Phase I, CP-US-302, Part I, 17 December 1976.
- (4-8) Computer Program Product Specification for the GPS X User Set (Aided), CP-US-302, Part II, 11 March 1977.
- (4-9) GPS User Equipment (UE) Orientation Course, Student Handbook, MX-125-C-US-7701, Magnavox APD, Torrence, CA.
- (4-10) Navigation Software Design for the User Segment of the NAVSTAR GPS, D. W. Klein, et al, Intermetrics Inc., Cambridge, MA.
- (4-11) Prime Item Product Function Specification for the User Equipment Set X of the NAVSTAR GPS User Equipment Segment Phase I, CID-US-121, 14 January 1977.
- (4-12) Computer Program Development Specification for the Set X Signal Processing Software (X-SPS) of the NAVSTAR GPS User Equipment Segment Phase I, CP-US-300, Code Ident. 12813, Magnavox APD, 7 October 1977.

- (4-13) System Specification for the NAVSTAR Global Positioning System, Phase II, SS-GPS-200, 1 September 1977.
- (4-14) System Segment Specification for the User System Segment NAVSTAR Global Positioning System, Phase II, SS-US-200, 1 September 1977.
- (4-15) L.L. Horowitz and S. R. Splar, ECM Vulnerability of the GPS User Receiver in a Tactical Environment, PR-XR1, MIT Lincoln Laboratory, 7 May 1976.
- (4-16) W.M. Stonestreet, "A Functional Description of the NAVSTAR GPS Receiver Model-X", CSDL Report (for SAMSO) R-981, 26 April 1976.



## APPENDIX 6-B

### FUNCTIONAL DESCRIPTION OF THE JTIDS CLASS-II TERMINAL

Draper Laboratory was recently involved in an effort\* that required the functional definition of a TDMA Class-II JTIDS terminal. This terminal had to be capable of performing the composite functions of the Hughes Improved Terminal (HIT) and the Singer-Kearfott Division Class-II terminal. In addition to the JTIDS communications and relative navigation functions, the terminal also processed TACAN and IFF signals. This functional definition was approved by representatives of the JTIDS Joint Program Office (JPO). Subsequently, the JPO has developed specifications for a Phase-2 Advanced TDMA Class-II terminal. The major differences are that the IFF signal-processing requirement was eliminated and the advanced TDMA signal structure was chosen. The changes on signal structure have impact on the signal- and data-processing portions of the terminal. However, it is felt that the functional terminal definition developed by CSDL will be useful to some readers, and has been reprinted as an appendix to Section 6. Classified portions have been omitted.

---

\* Stonestreet, William M, et al, GPS/JTIDS/IMS Integration Study Final Report, CSDL Report R-1151, May 1978.

### 5.1 Functional Description

The functional block diagram of the JTIDS Composite-Baseline Set is shown in Figure 5-1. The major functions are provided by the antennas, antenna interface unit, RF power amplifier, transceiver, signal processing unit and data processing unit.

The transmitted signal is routed from the transmitter through the RF power amplifier and antenna interface unit to the antennas. During transmission, Cyclic Code Shift Keyed (CCSK) data is Minimum Phase Shift Keyed (MPSK) onto a frequency hopped LO, and then up-converted using the same local-oscillator signals as the receiver channels. The receiver signal from the antenna is down-converted twice with a fixed first LO and a frequency-hopped second LO. Eight parallel receiver channels are used for preamble detection, and one channel is used as a data channel. The second LO frequency for each channel is developed from one of the eight synthesizers which are controlled from the signal processor, as described above for transmit.

The signal processing unit and data processing unit perform the basic message formatting and terminal synchronization. The secure data unit works with the two processors for data encryption and decryption.

During transmission and reception, the interface between the analog (RF/IF) and digital subsystems is via the signal processing unit with digital data routed through the secure data unit. The data processing unit provides the I/O interfaces and the I/O multiplexer bus for interfacing with other auxiliary or peripheral devices and the central computer on the aircraft.

The data processing unit performs several other important functions, including (1) coordinate conversion of received position data, (2) interfacing with the signal processor and units outside the terminal, (3) control of net processing and time synchronization, (4) operator interface for the Control and Display Panel, and (5) message reformatting.



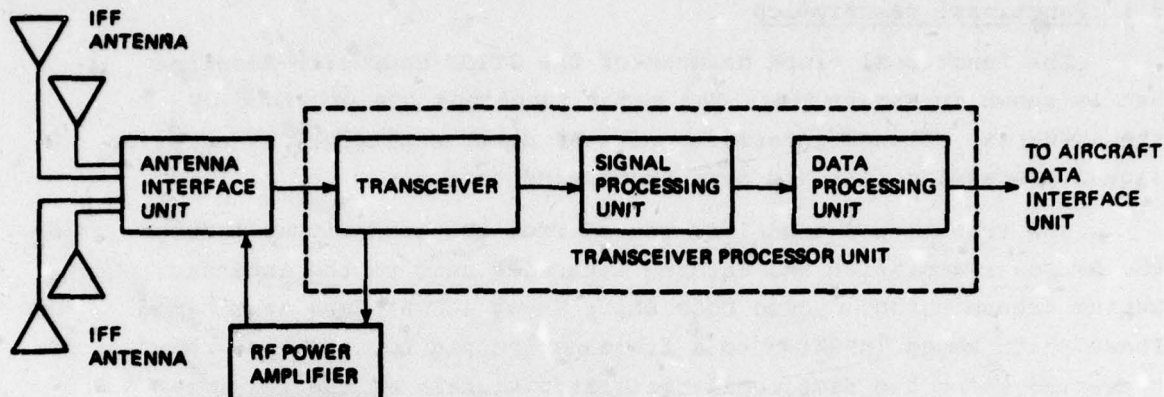


Figure 5-1. Functional block diagram - JTIDS composite baseline set.

#### 5.1.1 Antennas and Antenna Interface Unit

The recommended JTIDS/TACAN/IFF antennas are the TACAN and IFF antennas already existing on the subject aircraft. For a more detailed discussion on antennas see Section 13.

The Antenna Interface Unit provides interconnection between the transmitter, receiver, and these four antennas (see Figure 5-2). The output of the transmitter RF Power Amplifier is passed through one of two bandpass filters. A wide-band (960 to 1215 MHz) filter is used for TACAN, IFF, or JTIDS Mode 1 (frequency hopped). For JTIDS Modes 2, 3, or 4 (non-hopped modes), a narrow-band filter at 969 MHz is switched in. Following the bandpass filters are band-reject filters at 1030 and 1090 MHz used in the TACAN mode or any JTIDS mode. These filters are to prevent an inadvertent transmission at the IFF frequencies. These filters are switched out (bypassed) for the IFF mode of operation.

A simple ferrite circulator is used as the T/R device. The circulator is followed by a dual directional coupler to detect the forward and reverse power levels to and from the antennas. The detected levels are used for self test and for transmitter protective shutdown in the event of excessive reflected power. A low-pass filter is included to further reduce the transmitter harmonic power levels. The switch which selects the upper or lower antenna is also included in this unit.

A three-way power combiner sums the outputs of two (upper and lower) IFF antennas with the JTIDS/TACAN received signal from the circulator. In the receive path, a three-position switch selects either

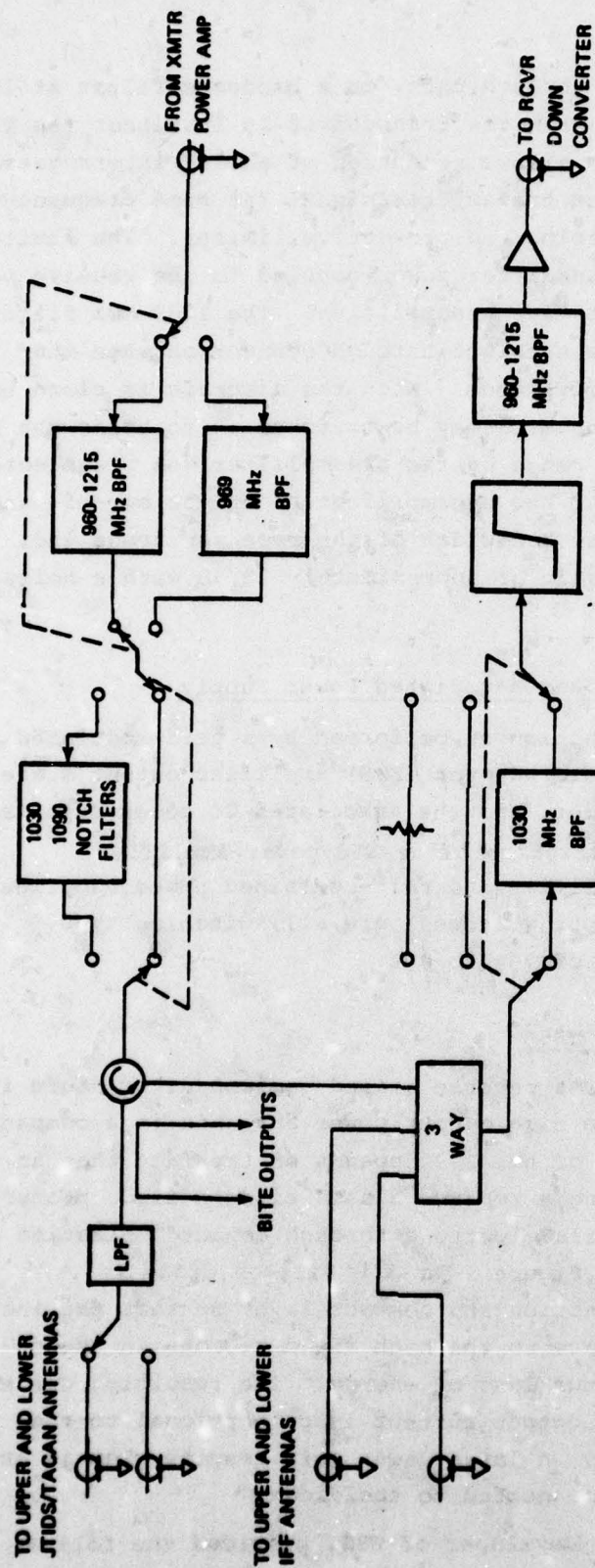


Figure 5-2. Block Diagram—Antenna Interface Unit—Baseline-P/  
Alternative I/Alternative II



an attenuator, a straight through path, or a bandpass filter at 1030 MHz. The filter is switched in when the transceiver is in either the TACAN or JTIDS transmit mode. This allows reception of an IFF interrogation at 1030 MHz while blocking the transmitter signal (at some frequency other than 1030 MHz) from triggering the protective limiter. The limiter is required to reduce the transmitter power coupled to the receive path to prevent damage to the sensitive preamplifier. The 1030 MHz filter is normally bypassed with the straight-through connection when the transceiver is in the receive mode. When the aircraft is close to a TACAN transponder, an attenuator may be switched in to bring the received signal down to the linear range of the preamplifier. A preselector filter is provided ahead of the preamplifier to reject out-of-band signals and limit the noise bandwidth of the receiver front end. The preamplifier will have a gain of approximately 23 dB with a noise figure better than 3.5 dB.

#### 5.1.2 RF Power Amplifier and Associated Power Supply

The RF amplifier function is performed by a grid-modulated Electronic Bombardment Semiconductor (EBS) amplifier output stage, a transistor input amplifier, and the associated DC power supplies. Table 5-1 provides a description of an EBS power amplifier, with transistor input amplifier and self-contained power supplies. The power supplies (4 output voltages) are all switching type, with typical efficiencies of 70% to 80%.

##### 5.1.2.1 Operation of the EBS\*

Impact ionization of a reverse biased semiconductor diode is used in the EBS to achieve high output power and gain in a compact, rugged device. Operation of the EBS depends on the fact that an energetic electron striking a reverse biased silicon diode produces over 2,000 additional mobile electrons through impact ionization of the silicon. As shown in Figure 5-3a the silicon diode is fabricated with a thin junction and contact layer so that the incident electrons can penetrate to the high field regions in the depleted junction with minimum loss of energy. The resulting current gain is linear, since the output current is proportional to the incident electron current. A large power gain results when an external resistive load is connected to the diode.

\*Watkins-Johnson Company, developer of EBS, provided the following text and Figure 5-3.

A simple grid modulated EBS RF amplifier can be designed as shown in Figure 5-3b and consists of a cathode, grid, semiconductor diode and input/output circuitry. An input signal applied between the grid and cathode modulates the electron beam which then strikes the semiconductor diode. The electron beam can be controlled with a low level RF signal by using a fine mesh grid located close to the cathode. The cathode bias supply is chosen so that if no input signal is applied, no electron beam current flows. Thus the output power can be modulated solely by the input RF signal. Output matching, typically a section of microstrip transmission line, is included to electrically match the semiconductor diode at the desired operating frequencies.

Table 5-1. EBS RF power amplifier characteristics

Bandwidth	960-1215 MHz
Output Power	
Average	125 W (+51 dBm)
50% duty cycle	250 W
1% duty cycle	750 W
Output Impedance	50 $\Omega$
Output VSWR	1.5:1 (MAX)
Gain (EBS)	23 dB
Efficiency (EBS)	40% (MIN)
Input signal level (EBS) <sub>AVG</sub>	+28 dBm
Gain (driver amp)	28 dB
Input signal level (driver amp) <sub>AVG</sub>	0 dBm
Input impedance (driver amp)	50 $\Omega$
Input VSWR (driver amp)	1.5:1 (MAX)
Cooling	Forced Air
Bias Voltages	
Heater	6.3 V
Grid Bias	8 V
Electron Beam	12k V
Diode bias, driver amp	80 V
Estimated Volume	0.23 ft <sup>3</sup>



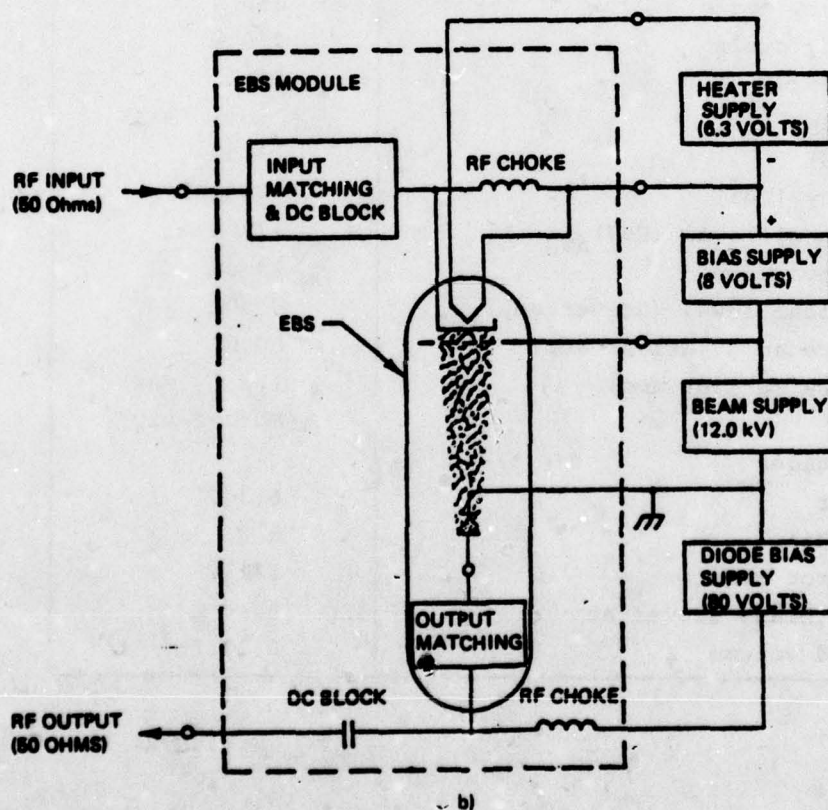
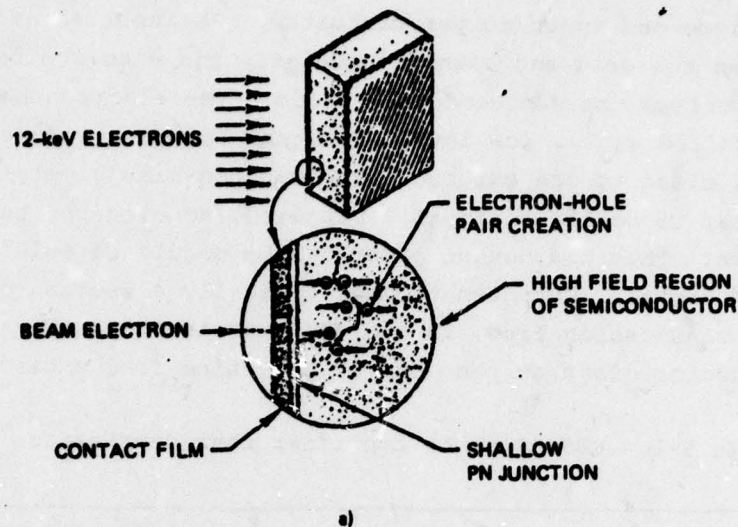


Figure 5-3. EBS power amplifier.

### 5.1.3 Transceiver

#### 5.1.3.1 Up/Down Converter

The up/down converter, as shown in Figure 5-4, transforms the received RF signal to the 360 to 615 MHz first IF band and transforms the transmitted IF signals to the transmitted RF band. The dehoppping of the receive signal is accomplished in the next assembly, the detector, and as such, only a single conversion step using a fixed 1575-MHz LO is provided in the down converter. Since each detector module contains two identical detectors, the down converter provides only four outputs, one to each detector module. A fifth output is provided for the IFF reception function wherein the 1030 MHz received IFF signal is converted to 545 MHz by the down converter. Since the receiver preamplifier is located in the Antenna Interface Unit(AIU), no additional preamplification is provided in the down converter. A preselector filter is provided to eliminate undesired out-of-band signals that may have been introduced in the cable run from the AIU to the transceiver and to provide additional filtering to the signals received by the antennas. Appropriate first IF amplification and filtering are provided in the down converter.

The up-converter input contains a switch to select one of three input signals. For JTIDS, the 315-MHz output from the modulator is selected. For TACAN, the dithered LO (251, 252, or 253 MHz) is selected. The dithering is accomplished in the comb generator module. The third selectable input to the up converter is a fixed 256-MHz CW signal from the comb generator module. This signal is used to generate the 1090-MHz IFF transmitted signal CW source.

The selected input to the up converter is mixed up to an 1826 to 1892-MHz band using the same 1575-MHz LO as used in the down converter. The signal is then mixed down to the 960 to 1215 MHz transmit band using the channel-one output of the comb filter/output module for an LO. This LO is hopped for JTIDS and falls in the 675 to 930 MHz range at 3-MHz increments. For TACAN, the LO is selectable from 677 to 802 MHz at 1-MHz increments and the LO is 741 MHz for the IFF function. This yields a transmitted RF range of 1025 to 1150 MHz at 1 MHz steps for TACAN and a fixed 1090 MHz for the IFF function. Appropriate amplification and filtering is provided at each up converter IF and RF stage.



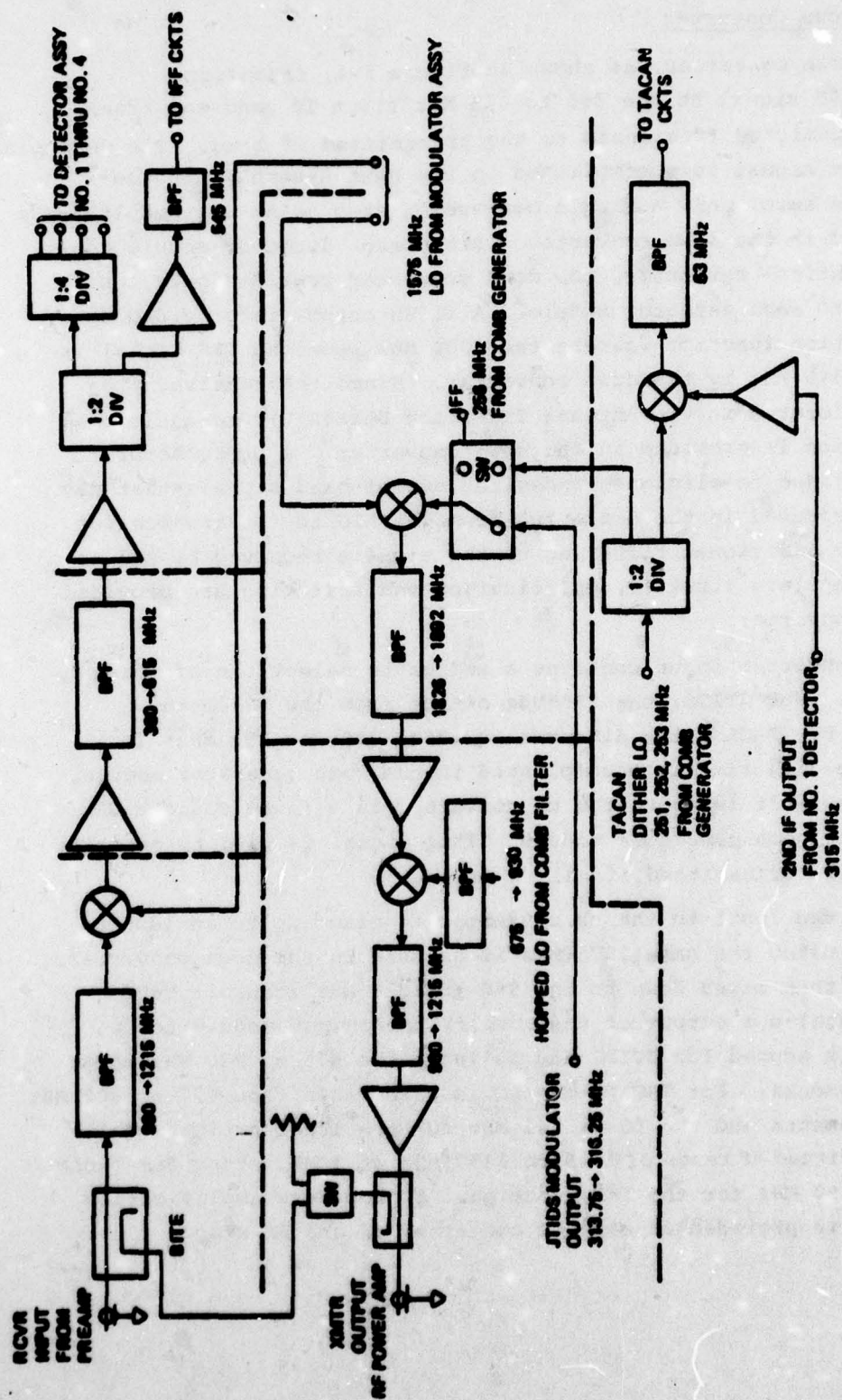


Figure 5-4. Block diagram Up/Down Converter Baseline and Integrated Design I

To perform short RF loop self test during transmission, a BITE coupler is provided at the up converter output and the down converter input. A switch connecting the two couplers is provided to perform the short RF loop test. Long RF loop test is provided by the AIU. Comparison of the short and long loop tests will isolate which LRU is faulty in the event of a hard or soft failure.

The up/down converter also contains the final TACAN down converter stage wherein a 315 MHz second IF output from one of the eight detectors is mixed with the dithered LO (251, 252, or 253 MHz) to yield the receiver final IF of 63 MHz as an input to the TACAN demodulators. The center frequency, 252 MHz, of the dithered LO is chosen as being 63 MHz below the receiver second IF of 315 MHz. The TACAN transponder on the ground or on another aircraft (air-to-air TACAN mode) converts the signal it receives to two frequencies, one 63 MHz above what it received and the second 63 MHz below and sends back the two converted frequencies. (The received frequency 63 MHz below what was transmitted is not used by this transceiver.) Using the same up and down converter LO's in the transceiver, the TACAN received frequency (63 MHz above what was transmitted by the transceiver) will be down converted to a frequency 63 MHz above the up converter input frequency of  $253 \pm 1$  MHz. In this manner, the same receiver second IF of 315 MHz may be used for both JTIDS and TACAN and one detector channel may be shared. Coupling off that detector channel at 315 MHz and mixing with the dithered LO in the up/down-converter module the TACAN reply is dithered. The dithering is used so that the transceiver may distinguish its reply from the TACAN transponder from that of other aircraft. Each transceiver will have a statistically unique or purely random dithering control.

In both the TACAN and IFF modes, the output from the up converter is a CW signal at the desired RF frequency. Modulators in the power amplifier assembly provide the proper TACAN or IFF signal modulation. When operating in the JTIDS mode, the up-converter input is already modulated (see Section 5.1.3.4) and, as such, no further modulation is required. The RF power amplifier (see Section 5.1.2) acts strictly as a power amplifier for JTIDS.

#### 5.1.3.2 Frequency Synthesizer

The basic synthesizer approach using SAW filter banks is discussed in Section 14.7. The synthesizer consists of three modules, one comb



generator module and two identical comb filter/output modules. The comb generator module uses a 1-MHz input from the modulator module to generate two comb spectra and two fixed LO's used by the comb filter/output modules. In addition, the comb generator also generates the TACAN dithered LO and the IFF transmitted 256 MHz frequency as discussed in the previous section. The comb generator is shown in the block diagram of Figure 5-5.

Each comb filter/output module generates four independent LO signals (see Figure 5-6). By selecting one frequency from each of the two comb spectra and mixing these two frequencies with one or two fixed LO's, any frequency in 3-MHz steps in the 675 to 930 MHz range may be generated. This covers the requirements for the hopped LO's for JTIDS. The eight outputs from both of the comb filter/output modules thus provide the hopped LO for each of the eight detectors. The output from channel one of one of the two comb filter/output modules is also used for the up converter hopped LO.

Since one of the fixed LO's used in channel one on one of the two comb filter/output can be varied plus and minus 1 MHz by a switch on the comb generator module, a portion of the band, 677 to 802 MHz, can be used for the TACAN 1 MHz channel spacing requirement. Thus this LO channel is used for the receiver and transmitter TACAN channel selection. In addition, this LO channel, since it is the only one tied to the up converter and since it can generate 741 MHz, will be used for generation of the 1090 MHz IFF responses as discussed in the previous section.

#### 5.1.3.3 Reference Oscillator

The basic 10 MHz clock oscillator used to generate all LO's and system timing clocks is discussed in Section 4.1.2.2.

#### 5.1.3.4 Modulator

The modulator module generates several fixed LO signals, as well as MPSK modulating the JTIDS up converter input signal. The block diagram is shown in Figure 5-7. A SAW oscillator is used to generate 313.75 MHz which is used by both the MPSK modulator on this module as well as by the two MPSK demodulators on each of the four detector modules.

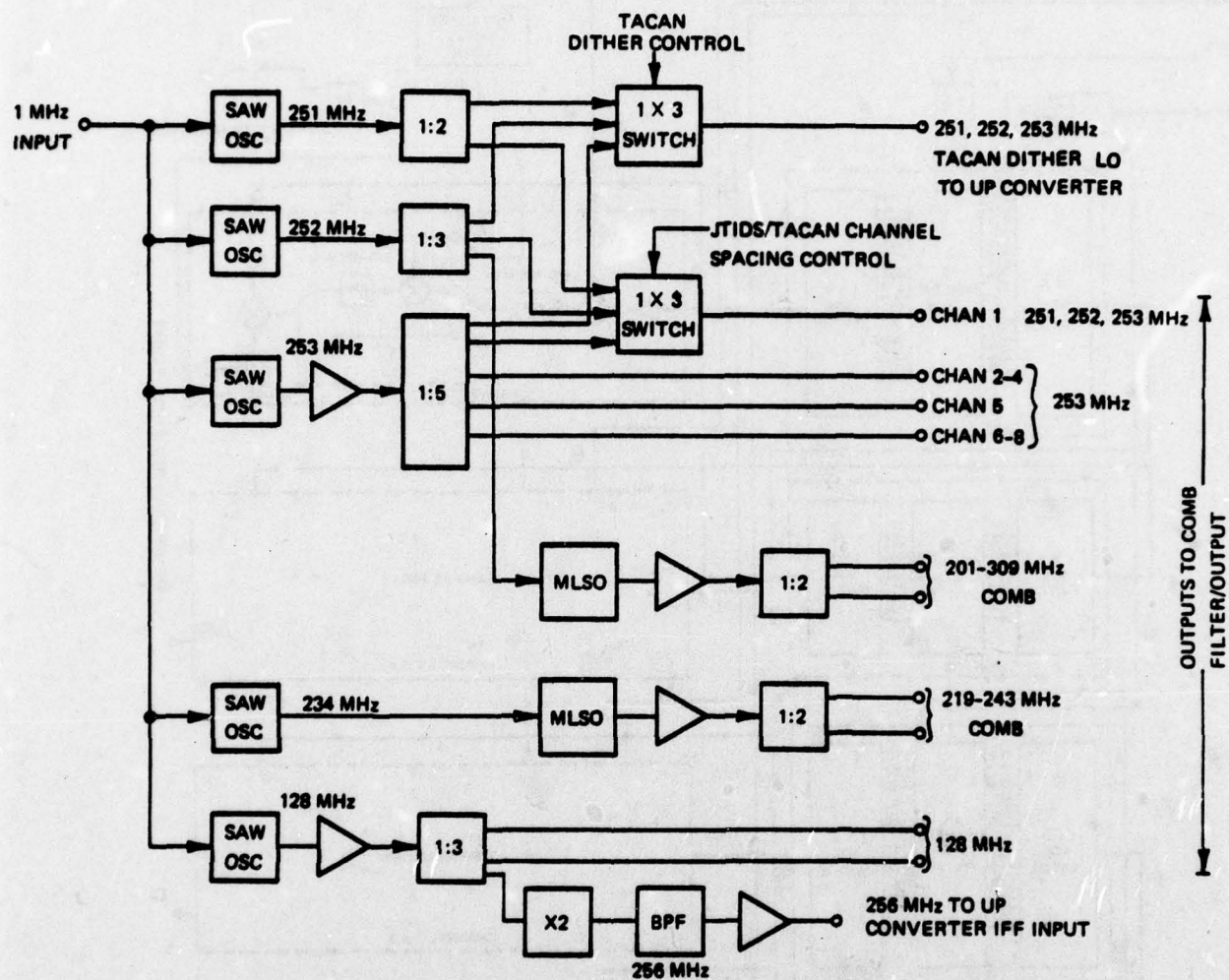


Figure 5-5. Block diagram SAW Oscillator/Comb Generator Baseline and Integrated Design I.



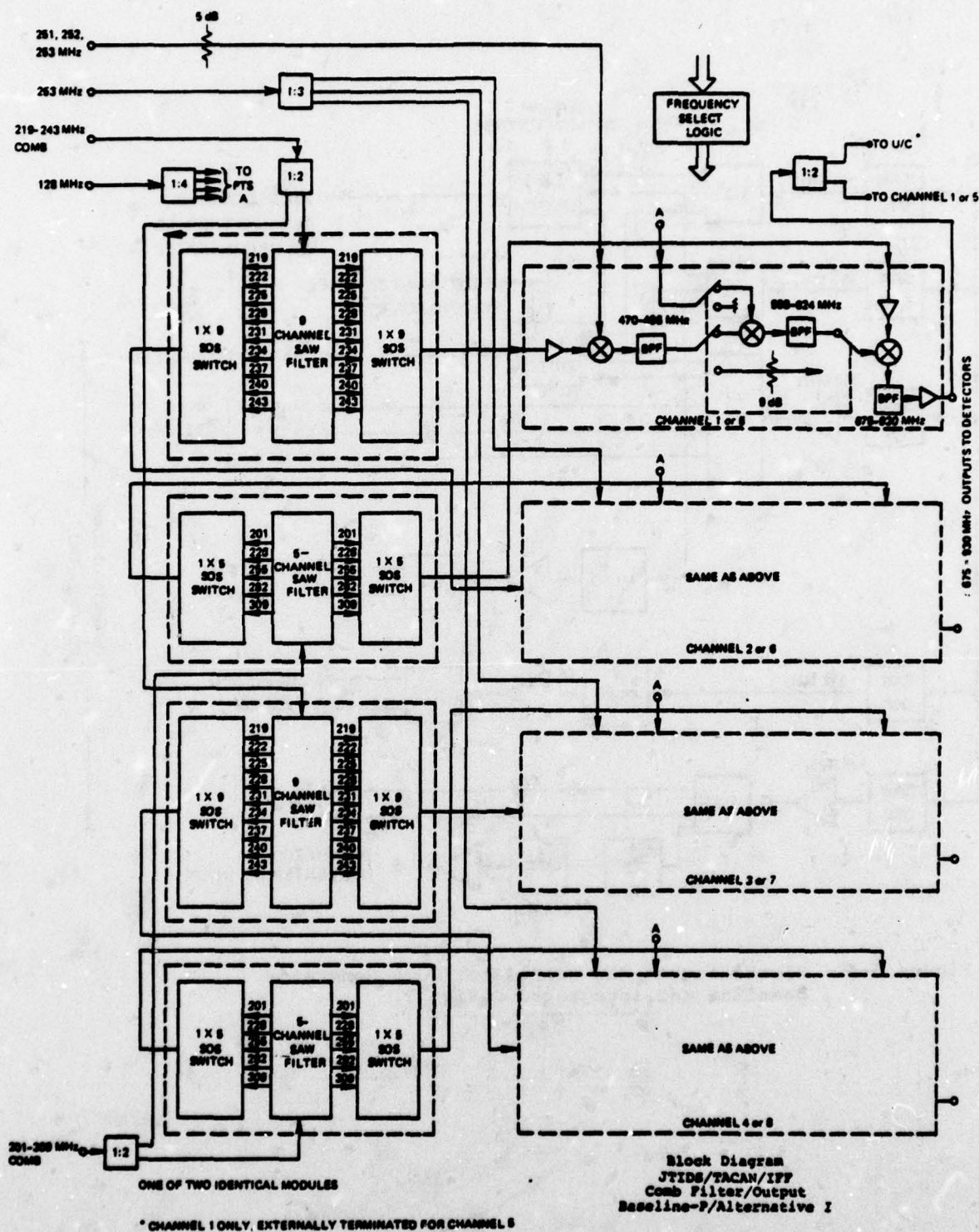


Figure 5-6. Block diagram—Comb Filter/Output Baseline and Integrated Design I

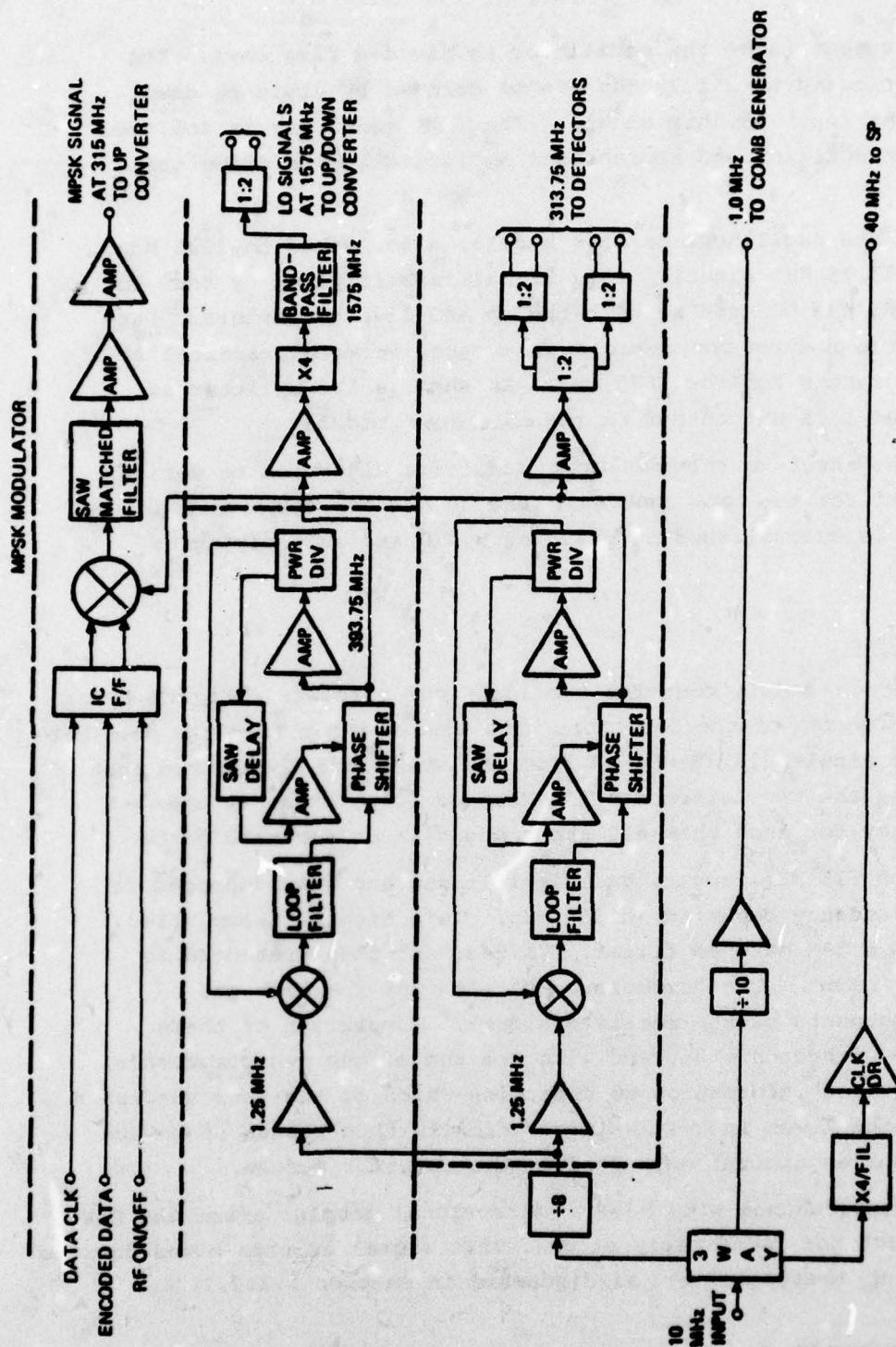


Figure 5-7. Block diagram—modulator module—baseline and Integrated Design I.



As such, the output of the oscillator is divided five ways. The oscillator is locked to a 1.25 MHz signal derived by dividing down from the 10 MHz input to this module. The MPSK modulator is followed by a SAW matched filter and appropriate amplification to drive the up converter.

A second SAW oscillator on this module, also locked to 1.25 MHz, generates a 393.75 MHz signal. This signal is multiplied by four to obtain the 1575 MHz LO used by both the up and down converters. Partitioning of the up/down converter module makes it more practical to have separate inputs for the 1575 MHz. As such, a 1:2 splitter is provided at the 1575 MHz output on the modulator module.

The 10 MHz input to the modulator module is also used to derive the 1 MHz input for the comb generator and 40 MHz for the signal processor. This is accomplished by dividing by 10 and multiplying by 4 respectively.

#### 5.1.3.5 Detector

Each detector module contains two identical circuits. Figure 5-8 shows a block diagram of the detector. The single input from the down converter and the single 313.75 MHz LO from the modulator module are each split to supply the two detectors. The hopped LO's, being at a different frequency for each channel, are brought in on separate paths.

The 360 to 615 MHz input signal is filtered and then dehopped to yield an IF frequency centered at 315 MHz. This signal is amplified, passed through a SAW matched filter, limited, and then presented to the MPSK demodulator. The demodulator detects the in-phase and quadrature components of the received signal. Comparison of these two components with each other and with the sum of the two components yields the required information to determine which of the four quadrants the received signal was in. This information is then passed on to the signal processor as digital outputs from the detector module.

One detector channel will have a directional coupler after the SAW filter to extract the TACAN reply signal. This signal is then mixed down to 63 MHz in the up/down converter as discussed in Section 5.1.3.1.

#### 5.1.3.6 Power Supply

The estimated total dc power required for this equipment is 750 watts; thus 375 in<sup>3</sup> is also required. A description of the multivoltage switching power supplies recommended for the equipment is given in Section 14.4.

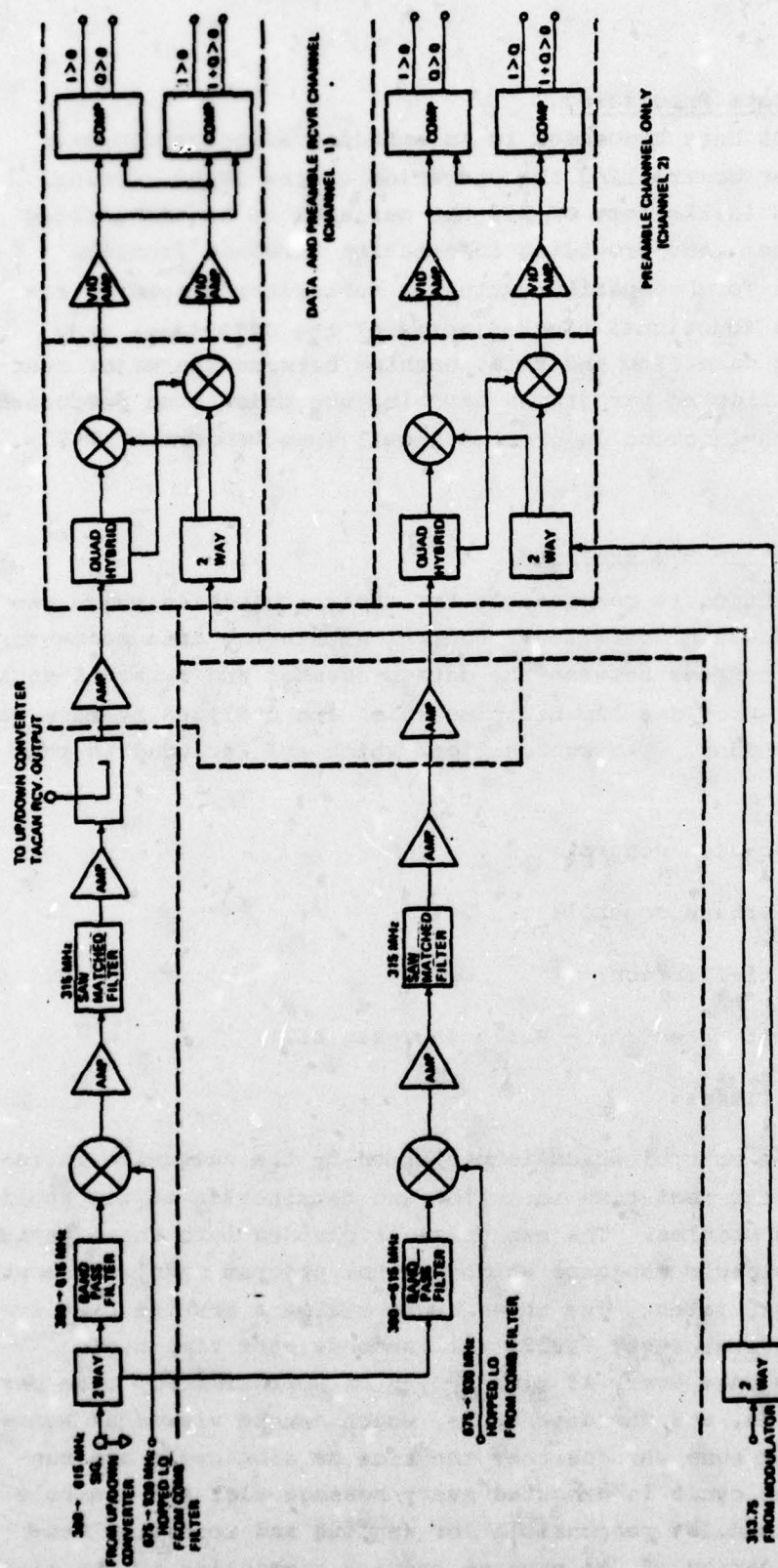


Figure 5-8. Block diagram—dual detector module-baseline and integrated Design I.



#### 5.1.4 JTIDS Data Processing

The JTIDS Data Processor is an embedded avionics computer responsible for controlling the operation of the JTIDS terminal, providing data in the form of 233-bit messages to be transmitted to the JTIDS net, and providing information received from the JTIDS net in a form compatible with the subscriber's needs. Figure 5-9) is a functional block-diagram of the JTIDS data processor showing data flow and relationships between the major functions. The following paragraphs describe the processing performed as part of each function in greater detail (see References 5-1 through 5-13).

##### 5.1.4.1 Executive and Services

This function is responsible for those activities which establish and maintain operational control within the data processor and at the interfaces between the data processor and external equipment. It also provides capabilities which are utilized by many of the other functions. The subfunctions which are included in this area are:

- (1) execution control;
- (2) interface control;
- (3) initialization;
- (4) fault detection - Built In Test (BIT);
- (5) utilities.

Execution control which is performed by the executive is responsible for the real-time initiation and termination of all application-program modules. The executive is divided into three parts referred to as cycle managers which control program modules executing at different rates. The three cycle managers are the slot cycle, which executes every 7.8125 milliseconds, the time cycle, which executes once every 13 slot-cycles or approximately once per 100 milliseconds, and the data cycle, which can be viewed as background since it runs when neither the time or slot cycle are running. The slot cycle is executed every message slot and controls those program modules responsible for sending and receiving messages. The majority of the program modules controlled by the slot-

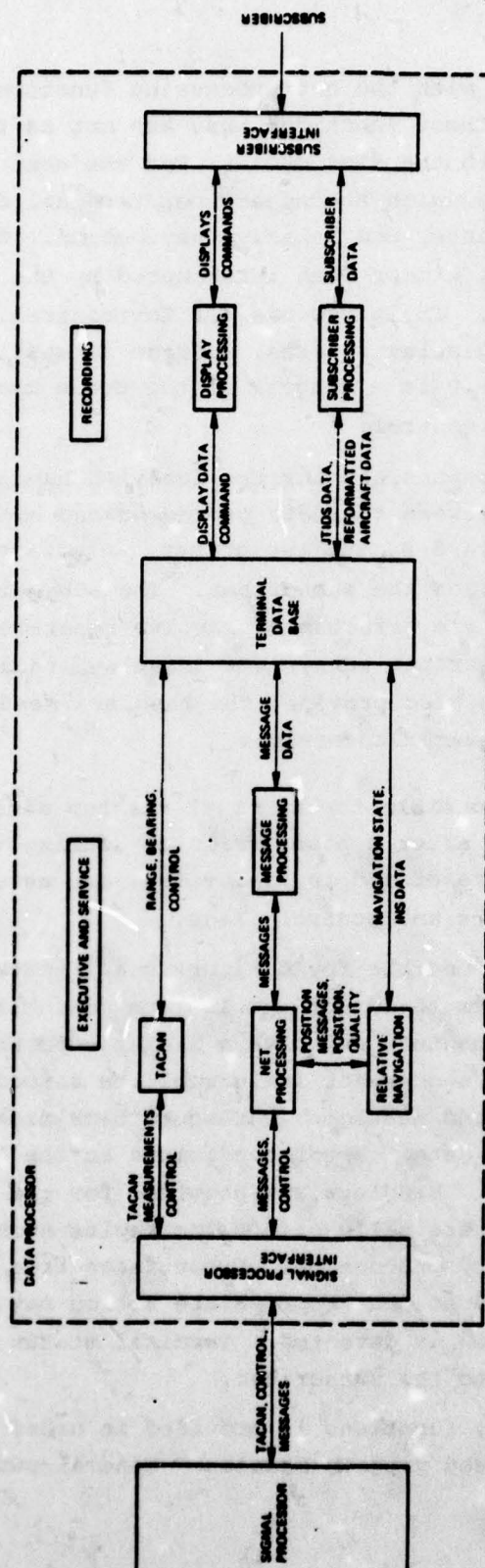


Figure 5-9. JTIDS data processing.



cycle manager are associated with the net-processing function. The time-cycle manager controls those functions that are not as time critical as those associated with the slot cycle. For the most part, these functions are associated with the subscriber/terminal data transfers, data-base maintenance, and relative navigation. The data cycle is always running, except when interrupted by the higher-priority slot or time cycles. It is responsible for controlling the low-rate processing such as display updates, message formatting, and clock maintenance. Figure 5-10 is a diagram of the cycle managers and the functions which they control.

Interface control is responsible for the detailed handshaking necessary to transfer data between the data processor and external equipment. As shown in Figure 5-9, the two primary interfaces involve the signal processor and the subscriber. The subscriber interface could utilize a single data bus or involve separate channels for each of the subscriber subsystems (display, radar, weapons, etc). This function also provides the handlers needed to respond to externally generated interrupts.

Initialization is responsible for terminal startup after power on or reinitialization after a power failure. During this process, all interrupts are reset and initial values are established for data-base variables and control flags.

Fault detection is responsible for monitoring all status words and error flags in order to detect and isolate potential error conditions. All I/O channels generate a status word which is examined to determine the success or failure of the associated data transfer. All transmitted messages are looped back from the signal processor in order to detect error conditions in the transmission/reception data paths. Handlers are provided for the internal interrupts which indicate malfunctions for faults such as arithmetic overflow and memory protection. Error flags from all program modules are monitored so that appropriate action may be taken when a problem situation is detected. Terminal status is continuously made available to the subscriber.

A single set of utility functions is provided in order to avoid duplication of often used program modules. General-purpose

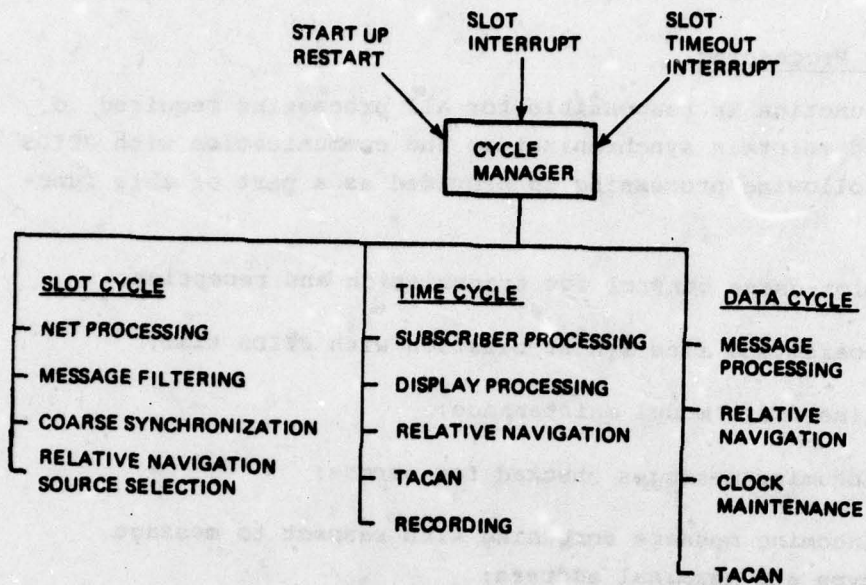


Figure 5-10. Cycle managers.



capabilities such as matrix manipulations and system inhibits and enables are maintained as part of this subfunction.

#### 5.1.4.2 Net Processing

This function is responsible for all processing required to establish and maintain synchronization and communication with JTIDS nets. The following processing is provided as a part of this function:

- (1) slot-usage control for transmission and reception;
- (2) coarse and fine synchronization with JTIDS time;
- (3) time drift model maintenance;
- (4) incoming messages checked for errors;
- (5) incoming message screening with respect to message type and terminal address;
- (6) error-free message routing to the message-processing function and/or relative navigation;
- (7) process time update (N4-1) and time-slot assignment (N3-1) messages;
- (8) process relay control (N2-1) messages;
- (9) machine acknowledge messages generation when required;
- (10) position extrapolation to reflect expected subscriber position at the time of position message (P1) transmission;
- (11) communication mode control (level of security) and operational mode control (normal, polling, radio silent, test).

The majority of the above processing is controlled by the slot-cycle manager and is performed one or two slots prior to, and two slots after, the transmission reception slot. Table 5-2 shows the slots in which various functions are performed relative to the slot in which the message is actually transmitted or received by the signal processor.

Table 5-2. Slot related functions.

N+2	N+1	N	N-1	N-2
Slot usage control and assignment	System mode check (radio silent, polling)	Message is received or transmitted (see Paragraph 5.1.4)	Received message signal processing (see Paragraph 5.1.4)	Message received in data processor
Relay output	Coarse sync processing			Message error check
Machine acknowledge output	Output message to Signal Processor			Message classification
Clock commands				Loopback error
Test message				Message routing
Position extrapolation				
Net control				



#### 5.1.4.3 Message Processing

This function receives complete messages from the net-processing function, unpacks and reformats the message data, and routes this data to the terminal data base, display-processing function, and/or subscriber-processing function as required. The category and subcategory of each message is examined to determine whether or not the message type is of interest to the subscriber. Unwanted messages are ignored. Incoming track and position messages are prioritized according to their range and range-rate. Because of limited storage space, only the highest-priority messages will be retained in memory. This information is also used to determine whether or not unknown or hostile aircraft pose an immediate threat to the subscriber. This function also collects the necessary data and formats outgoing messages based on control flags set by other functions.

#### 5.1.4.4 TACAN

This function is responsible for processing TACAN beacon signals in order to provide bearing and distance information. A total of ten Kalman-filter states are used to track the TACAN signal envelope, track the reference burst-pulses, and process beacon-range replies. The commands needed to control the TACAN related hardware with respect to frequency, pseudo-random beacon interrogation times and window widths, and signal acquisition are generated as part of this function.

#### 5.1.4.5 Relative Navigation<sup>(5-14,15,16)\*</sup>

This function provides subscribers position and velocity in both relative and geodetic coordinate frames. Input to this function consists of 3 JTIDS position messages (P1) that are screened for position and time qualities superior to the corresponding qualities within the terminal. Geometry also plays a role in selecting the three position messages along with their associated time-of-arrival which will be passed to the filter-processing portion of this function. Dead-reckoner inputs consisting of subscriber position, velocity, attitude, and timing data are also provided.

\* Superscript numerals refer to similarly numbered references in the List of References at the end of this section.

Filter processing provides a full-covariance Kalman filter utilizing the 15 states shown in Table 5-3. See Section 11 for a discussion of the relative navigation filter. The INS and clock-states estimate errors between the subscriber and the navigation- and time-controllers in the JTIDS net. The three position-messages are incorporated once every 16 seconds.

#### 5.1.4.6 Terminal Data Base

The terminal data base acts as a buffer between those functions responding to and controlling network-related activities and those functions responding to and controlling subscriber-related activities. It provides linkages and temporary storage for data which is passed between the various functions. It is responsible for the insertion of new data and the update or deletion of old data. It may also reduce the priority associated with track and position data, or extrapolate tracks and position data to reflect the passage of time. When information is required to continue processing, this function retrieves the data which satisfies the criteria specified as part of the request.

#### 5.1.4.7 Display Processing

This function is responsible for establishing and maintaining the primary man/machine interface for JTIDS. It performs the following major functions:

- (1) provides interactive communications with the subscriber;
- (2) creates displays and menus as requested by the operator;
- (3) processes command-received acknowledgements from the subscriber;
- (4) processes any subscriber-generated JTIDS messages;
- (5) retrieves and organizes JTIDS information according to the display mode and operator requests;



- (6) provides threat/hazard information immediately, even though it may not be within the current display range;
- (7) provides JTIDS received-command messages to the display in alphanumeric format immediately upon receipt.

Table 5-3. JTIDS filter states.

FILTER STATES	
Geographic position	3
Geographic velocity	2
INS tilts (relative)	3
Grid-relative position	2
Grid azimuth (wrt north)	1
Clock offset (relative)	1
Clock drift (relative)	1
Correlated velocity errors	<u>2</u>
Total	15

#### 5.1.4.8 Subscriber Processing

The primary responsibility for this function is to resolve differences and conflicts between JTIDS terminal-data formats and scaling and subscriber-data formats and scaling. JTIDS data is maintained within the terminal in a format which is compatible with the JTIDS message formats as specified in the Interim JTIDS Message Specification. Since there is little or no commonality between potential JTIDS subscribers, subscriber data is expected to occur in a wide variety of formats. The subscriber processing function tailors the JTIDS terminal to the existing subscriber information distribution system to the extent possible in order to minimize required subscriber modifications needed to accommodate JTIDS.

This function is also responsible for the processing that provides JTIDS information to various subscriber subsystems and vice versa. These capabilities are listed below:

- (1) provides subscriber-selected track-data to subscriber weapons-system to eliminate or reduce necessary radar scan for target lock on;
- (2) activates TEWS when threat is detected;
- (3) passes subscriber-designated track-data to central computer to provide HUD indicators;
- (4) correlate weapons-system radar-tracks with JTIDS tracks in order to identify weapons-system track on JTIDS display and report subscriber-target status to JTIDS net.

#### 5.1.4.9 Recording

This function provides information to a recording device so that terminal processing and information flow within the terminal can be examined at a later time.



#### 5.1.5 Processing Requirements

As mentioned in previous paragraphs, the JTIDS baseline is a composite drawn from existing terminals, preliminary designs and emerging system requirements. Thus, in order to establish estimates for memory requirements and processing loads, a number of assumptions concerning the JTIDS software design and functional requirements were made. These assumptions are outlined below:

- (1) The messages to be implemented are listed in Table 5-4.
- (2) The terminal data-base will maintain track-data relative to two points, the subscriber and a variable-center controlled by the subscriber.
- (3) Incoming track-data is prioritized on range and range-rate which is also used for threat assessment.
- (4) The track data-base is updated with new track information based on track System Reference Number (SRN) and calculated priority.
- (5) Subscriber-selected track-data may be routed to the weapons system to assist radar acquisition of targets and to a central computer for HUD use. Once selected by the subscriber, updated track data will be sent to the weapons system or the central computer automatically as new tracks are received.
- (6) The display is extrapolated and refreshed ten times-per-second.
- (7) The priority of each track in the data base is lowered twice a second to reflect the passage of time. The track is deleted when the priority reaches zero.
- (8) Subscriber radar data is received by the terminal and correlated with current track data so that current targets can be reported to the JTIDS net.
- (9) Display switch actions are limited to information-filter control, command acknowledge and special points input.

Table 5-4. JTIDS messages.

Message Code	Description	Use
T-1	Hostile/unknown air tracks	RCV
P-1	Own position	XMT
C-1-1/M3-1	Command/target	RCV
P-1	Friendly air track	RCV
T-1	Radar lock on hostile/unknown	XMT
I7-1	Special points-hostile SAM/AAA	XMT/RCV
V1-1	Tanker assignment/close vector	RCV
P-2/P-3	Friendly surface/ship	RCV
I1-1	Weather	RCV

The estimates found in the following paragraphs are based on the performance characteristics of the Hughes HMP1670 computer. The functional characteristics of this machine are given in Table 5-5. A more complete description of the M HP1670 is given in References 5-6 and 5-7.

#### 5.1.5.1 Memory Requirements

The estimated memory-size for the JTIDS composite-baseline broken down by function is given in Table 5-6. These estimates are based on implementation information gathered from the current Singer and Hughes terminal-development contracts along with coding estimates for those functions which have not been implemented (also see References 5-10, 11, 12, 17). All estimates assume that the software is coded in assembly language.

#### 5.1.5.2 Processing Loads

Table 5-7 gives the processing loads for each major function for the JTIDS composite-baseline. These loads are based on actual implementation times from Hughes and Singer as well as estimates for those functions which have not yet been implemented (see References 5-10, 11, 12, 17). The total estimated load is 103.4% which implies that the M HP1670 does not have enough processing power to handle the JTIDS functions as outlined in the preceding paragraphs. These processing-load



estimates are very sensitive to assumptions made regarding the software design and the methods of implementation. The estimated load for message processing assumes that the range, range-rate and priority calculations utilize fixed point and cordic instructions. If floating-point instructions were used and trigonometric subroutines replaced the cordic instructions, the message-processing load would increase from 18% to at least 35%. By the same token, if the subscriber were willing to tolerate a 10 to 15 second delay while the display was filled with incoming track data, the need to maintain track-data for more than one display-center could be eliminated. This would reduce the message-processing load by 8 to 9% and the data-base processing-load by about 4%.

The throughput in terms of KOPS (thousands of operations-per-second) for the HMP1670 is calculated for two avionics-instruction mixes in Table 5-8. Table 5-9 gives the processing load in KOPS for the major functions in the JTIDS-composite baseline. These KOPS figures provide a measure of the JTIDS-processing load which is somewhat independent of the computer chosen for implementation and will be used in later paragraphs when sizing the integration alternatives.

Table 5-5. HMP-1670 functional characteristics.

Type	General purpose, stored program, parallel
Control	Microprogrammable
Microprogram memory	2048 36-bit words (expandable to 4096 36-bit words)
Micro-instruction execution time	200 nanoseconds
Emulation type	Interdata 70 and 7/16
Arithmetic	Two's complement, integer
Data word lengths	8, 16, 32, 48 and 64 bits
Data types	Fixed-point and floating-point
Data flow	16-bit parallel (halfwords)
ALU width	32-bit
Number of instructions	151
Instruction word length	16 and 32 bits
General registers	Sixteen 16-bit hardware registers and eight 48-bit floating-point memory registers
Storage (read/write)	Random access, dynamic, integrated circuit, limited non-volatility via battery backup
Storage (read only)	Programmable read-only, non-volatile
Storage size (maximum)	131,072 halfwords (17 bits each including one parity bit)
Addressable unit	8-bit byte
Storage cycle time	600 nanoseconds
Storage access time	400 nanoseconds
Memory protect resolution	128 halfwords
Program loadable counters	Two program loadable counters with selectable interrupt (one readable, incrementing, 32-bit counter) (one decrementing 16-bit counter)
Parallel I/O bus	I/O mux bus using programmed I/O, automatic I/O and interleaved data channel
DMA rate	1.6M halfwords/sec
Level of interrupts	8
External interrupts	Vectored or polled using 255 device addresses



Table 5-6. JTIDS processing loads for Hughes MHP1670.

Function	Processing Load (%)
Executive and Services	10
Net Processing	25.8
TACAN	6.0
Relative Navigation	15.6
Message Processing	18.0
Data Base Processing	8.0
Display Processing	12.0
Subscriber Processing	5.0
Recording	3.0
Data Base	----
TOTAL	103.4

Table 5-7. JTIDS memory requirements.

Function	Memory (8-bit bytes)
Executive and Services	6,500
Net Processing	17,000
TACAN	4,400
Relative Navigation	11,800
Message Processing	7,500
Data Base Processing	1,500
Display Processing	7,000
Subscriber Processing	2,000
Recording	1,000
Data Base	31,000
TOTAL	89,700

Table 5-8. HMP1670 processing power.

Instruction	Execution Time (μs)	kops	Standard Aviation Mix (%)	Weighted kops	DIS* Mix	Weighted kops
Load/store	2.2	454.5	45	204.5	36	163.6
Add/subtract	9.66	103.5	9	9.3	14	14.5
Multiply/divide	12.21	81.9	5	4.1	7	5.7
Shift	2.8	357.1	5	17.9	4	14.3
Logical	2.0	500.0	5	25.0	8	40.0
Test and Branch	1.8	555.6	30	166.7	30	166.7
I/O	5.0	200.0	1	2.0	1	2.0
TOTALS				429.5		406.8
* Digital Integrating Subsystem				Average = 418.2		

Table 5-9. JTIDS processing load in kops.

Function	Processing Load (kops)
Executive and Services	41.8
Net processing	107.9
TACAN	25.1
Relative Navigation	65.2
Message Processing	75.3
Data Base Processing	33.5
Display Processing	50.2
Subscriber Processing	20.9
Recording	12.6
Data Base	----
TOTAL	432.5



## SECTION 5

### LIST OF REFERENCES

- (5-1) System Segment Specification for JTIDS Class II Terminal (Draft) (U), DCB78S3000, MITRE Corp., 30 January 1978.
- (5-2) System Specification for Joint Tactical Information Distribution System (JTIDS) (U), DCB76S0000, Code Ident No. 50464, 5 May 1977.
- (5-3) System Segment Specification for HIT Communications Terminal Hughes Improved Terminal (HIT) Program (U), Code Ident 05896, 1 June 1977.
- (5-4) Prime Item Development Specification for Transceiver Processor Unit, Hughes Improved Terminal (HIT) Program (U), Code Ident 05896, 30 September 1977.
- (5-5) Computer Program Product Specification for ACE Communications Terminal Operational Computer Program (AOCP) (Draft), CG-10044, Hughes Aircraft Co.
- (5-6) Message Specification (Interim) for Joint Tactical Information Distribution System (JTIDS) (U), DCB Exhibit 7501, Rev. A, 30 April 1977.
- (5-7) Attachment A to ESD Contract F19628-77-C-0151, Technical Criteria for JTIDS/F-15A Test Integration Program, Electronic Systems Command, Hanscomb AFB, 5 August 1977.
- (5-8) Prime Item Product Function Specification for Stand Alone Advanced Communications Processor, PP 10078, Hughes Aircraft Co., 30 September 1977.

- (5-9) A Small Size Terminal for the JTIDS Communications Network, The Hughes Improved Terminal, FR77-16-346, Hughes Aircraft Co., March 1977.
- (5-10) Development Specification Computer Program for JTIDS (F-15A) Interface Processor Set (Preliminary), Y201A570A100, Singer-Kearfott.
- (5-11) Study to Define the Integration of JTIDS into Four F-15A Test Aircraft, Final Report for Period June 1977 - December 1977, McDonnell Douglas Corp., January 1978.
- (5-12) Feasibility of Integrating the Hughes Improved Terminal (HIT) into F-15A JTIDS Test Aircraft, Final Report for Period October 1977 - April 1978, Contract F19628-77-C-0151, McDonnell Douglas Corp.
- (5-13) Interface Digital Processing Functions JTIDS Class II Terminal, R-1143, McCoy and Miller, The Charles Stark Draper Laboratory, Inc., February 1978.
- (5-14) Computer Program Performance Specification for the DFM (U), SKD Y201A450 Revision B, Code Ident No. 88818, Singer-Kearfott Division, 24 June 1977.
- (5-15) Relative Navigation Performance Specification (Draft), 60P2-PS-76-01, Revision A, 4 January 1977.
- (5-16) Relative Navigation Algorithm Specification for Joint Tactical Integrated Data System (JTIDS) (Preliminary Draft), 60P2-PS-7604, 13 December 1976.
- (5-17) ACE OCP Critical Design Review - Viewgraphs, FR77-17-1003A, Hughes Aircraft Co., September 1977.